NPS ARCHIVE
1966
KRUMM, T.

# MINE DELIVERY MODEL: A COMPUTER SIMULATION

THEODORE G. KRUMM
and
STEPHEN V. SNYDER

LIBRARY
NAVAL POSTGRADUATE SCHOOL
MONTEREY, CALIF. 93940

Approved for Public Release;
Distribution Unlimited

Approved for Public Release:
Distribution Unlimited

# MINE DELIVERY MODEL:

# A COMPUTER SIMULATION

by

Theodore G. Krumm
Lieutenant, United States Navy
B.S., United States Naval Academy, 1959

and

Stephen V. Snyder
Lieutenant, United States Navy
B.S., United States Naval Academy, 1959

Submitted in partial fulfillment
for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

UNITED STATES NAVAL POSTGRADUATE SCHOOL

May 1966

NPS ARCHIVE
1966
KRUMM, T.

Thesis
K875
c.1

ABSTRACT

The Mine Delivery Model is a Monte Carlo computer simulation of the delivery of mines in detail by ships, submarines, and/or aircraft. Inputs are of such a nature as to permit the simulated execution of any specific mining plan and the outputs provide sufficient information for subsequent evaluation of the plan by a threat assessment model.

The model as programmed for the CDC 1604 Computer consists of five programs. The first two programs convert input data, the third and main program is the mine delivery simulation, the fourth program provides a detailed printout of both inputs and game results, i.e., the calculated parameters for each mine layed accompanied by selected statistics, and the fifth program provides statistics and graphs on the distributions of the mine positions.

Library
U. S. Naval Postgraduate School
Monterey, California

TABLE OF CONTENTS


| Section | | Page |
|---|---|---|
| 1. | Introduction | 7 |
| 2. | Overview of the Mine Delivery Model | 11 |
| 3. | MINDEL: The Mine Delivery Model | 15 |
| | Aircraft Sequence of Events | 21 |
| | Ship Sequence of Events | 29 |
| | Submarine Sequence of Events | 33 |
| 4. | Applications of the Mine Delivery Model | 40 |
| 5. | Conclusions and Recommendations | 57 |
| 6. | Bibliography | 58 |
| Appendix | | |
| I. | Program GRID | 60 |
| II. | Program SURFIT | 81 |
| III. | Bottom Type Coding and its Application | 107 |
| IV. | Program Listing, Variable Names, and Input Format for MINDEL | 110 |
| V. | Flowcharts for MINDEL | 165 |
| VI. | Aircraft Circular Probable Error | 270 |
| VII. | Navigational Error | 273 |
| VIII. | Ship Attrition | 277 |
| IX. | Submarine Detection/Attrition | 280 |
| X. | Self-Propelled Mine Failures | 284 |
| XI. | Program MINDEL1 | 287 |
| XII. | Program LSQRPLOT | 313 |
| XIII. | Sample Run | 327 |
| XIV. | Flowcharts for Programs GRID, SURFIT, and MINDEL1 | 355 |

LIST OF ILLUSTRATIONS


| Figure | | Page |
|---|---|---|
| 1. | Schematic of the Over-all Model Concept in Mine Warfare. | 8 |
| 2. | Flowchart of Basic Logic in the Mine Delivery Model. | 16 |
| 3. | Graphical Examples of the Leg Concept. | 19 |
| 4. | Intended Mine Positions. | 43 |
| 5. | Sample Run - Final Mine Positions Generated by MINDEL. | 44 |
| 6. | Sample Run - Final Mine Positions Generated by MINDEL. | 45 |
| 7. | Sample Run - Final Mine Positions Generated by MINDEL. | 46 |
| 8. | Mine Frequency Distribution on the X-axis for all Sample Runs. | 47 |
| 9. | Mine Frequency Distribution on the Y-axis for all Sample Runs. | 48 |
| 10. | Mine Cumulative Distribution on the X-axis for all Sample Runs. | 49 |
| 11. | Mine Cumulative Distribution on the Y-axis for all Sample Runs. | 50 |
| 12. | Least Squares Fit to Mine Cumulative Distribution on the X-axis for all Sample Runs. | 51 |
| 13. | Least Squares Fit to Mine Cumulative Distribution on the Y-axis for all Sample Runs. | 52 |
| 14. | Randomly Generated Minefield Utilizing Distributions on the Axes. | 54 |
| 15. | Aircraft of Ship Leg Description List. | 64 |
| 16. | Submarine Leg Description List. | 66 |
| 17. | Scale for Bottom Material. | 108 |

18. Flowchart Symbols. 166

19. Aircraft Altitude versus Circular Probable Error. 270

20. Triangular Density Function. 274

21. Ship Probability of Damage Function. 277

22. Damage Type Scale for Ships. 278

## 1. Introduction

The Mining Delivery Model is a Monte Carlo computer simulation of sea mine delivery by ships, aircraft, and/or submarines. The purpose of the model is to provide the U. S. Navy the capability of simulating any planned mining operation through the utilization of high speed digital computing equipment. The over-all model concept in mine warfare includes several phases from mine selection through threat assessment, and mine delivery is but one phase of this over-all concept (see figure 1, a schematic showing the several phases and their interrelationships).

In early 1965 the Office of the Assistant to the Chief of Naval Operations for War Gaming Matters (OP-06C) established the requirement for a mine warfare modeling effort and assigned the task to the Computational Analysis Laboratory at the Naval Weapons Laboratory (NWL), Dahlgren, Virginia. In the summer of 1965 the authors of this thesis had the opportunity to view the state of the mine warfare gaming project at NWL. At that time the minefield threat assessment phase of the model was well underway and the mine delivery phase was scheduled to be begun in the spring of 1966. The authors then proposed to undertake the design and building of the mine delivery phase as a thesis project. The topic was of immediate interest to us due to our background experience in the mine forces and because it presented an excellent opportunity to exercise the analysis and gaming techniques studied in the Operations Analysis Curriculum

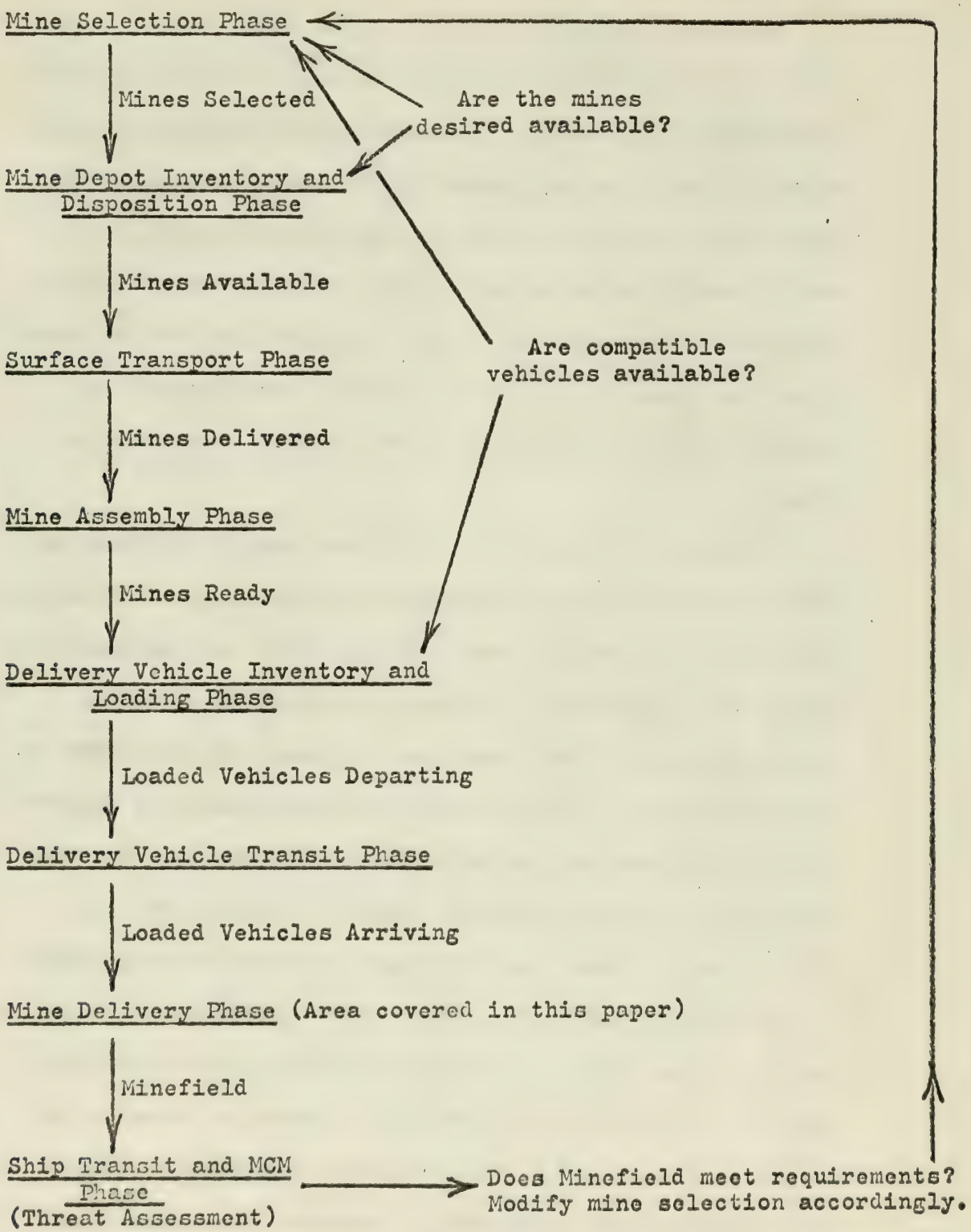



Figure 1. Schematic of the Over-all Model Concept in Mine Warfare

at the U. S. Naval Postgraduate School.

It has been the authors' objective that the mine delivery phase of the over-all model would provide the minefield data necessary for a tie-in operation with the existing threat assessment phase as developed by NWL. Every effort has been made to maintain liason with NWL to ensure compatibility of the model phases. Since several unique approaches to simulation problems have been included, special care has been taken to explain in detail each of these features in subsequent sections and to provide guidelines for the modification of the model where necessary to ensure flexibility and consistency with the requirements of the user.

At the outset it was decided that the model in this thesis would be unclassified, hence all data for the vehicles, mines, and the minefield used in the test runs shown in this thesis were arbitrarily or randomly generated and only represent a cross-section of authentic vehicles, mines and mining plans in the most general way.

The authors wish to take this opportunity to express their gratitude to Professor Alvin Andrus, Operations Analysis Department, U. S. Naval Postgraduate School, for assistance and encouragement as our thesis advisor; Lieutenant Colonel R. V. Schick, U.S.M.C., Office of the Assistant to the Chief of Naval Operations for War Gaming Matters, for his interest and contributions in co-ordinating our efforts with NWL's efforts; and to the personnel of the Amphibious Warfare Sec-

tion, Naval Weapons Laboratory, and the personnel of the Computer Facility, U. S. Naval Postgraduate School, for their generous cooperation.

## 2. An Overview of the Mine Delivery Model.

The Mine Delivery Model presented in this thesis simulates the delivery of sea mines by aircraft, ships, and submarines from the time of arrival of the delivery vehicle at the minefield objective area until it completes or aborts its mission, or is attrited, along with the determination of the pertinent mine parameters. The model is programmed in the FORTRAN 63 computer language for the Control Data Corporation 1604 Computer available at the U. S. Naval Postgraduate School; this is a comparable language to the FORTRAN IV language for IBM computers. Special care has been taken to ensure that the coding is also compatible with the IBM 7090/7094 series computers.

The following is a summary of the model's general characteristics.

The geographical limits are a rectangular area of any size which includes the minefield and all geographical points to be used in the simulation.

The offensive forces are limited to a total of 50 delivery vehicles and 167 mines. The vehicles are further limited to a maximum of ten aircraft types, ten ship types, and nine submarine types. Mines are also limited to a maximum of 24 aircraft types, ten ship types, and eight submarine types.

Enemy forces are simulated by appropriate vehicle attrition routines.

All the foregoing limitations may be increased with minor

program changes.

The structure of the Mine Delivery Model involves five computer programs linked by data storage on magnetic tape. The five programs are briefly described as follows.

Program GRID is an auxiliary program which transforms minefield geographical data in latitude and longitude to X and Y coordinates in yards relative to a pre-selected reference point. The converted data is stored on magnetic tape for use in program MINDEL. A detailed description of this program appears in Appendix I.

Program SURFIT is an auxiliary program which fits an Nth degree polynomial to a surface. In this particular application it is used to fit polynomials to (1) the minefield bottom contour and (2) the minefield bottom type surface. The coefficients and degrees of the two fitting polynomials are stored on magnetic tape along with the data from program GRID for use in program MINDEL. A detailed description of this program appears in Appendix II.

Program MINDEL is the main program which simulates the delivery operation. It is discussed in detail in the following section. It suffices to summarize here that program MINDEL utilizes the data from magnetic tape and punched card input data to game the events occurring to delivery vehicles and to determine the mine parameters according to the respective mine sequences. A printed summary narrative presents the details of vehicle events and the status of each mine. All input data and the results of each mine lay are stored on magnetic tape for use in program MINDEL1.

Program MINDEL1 is an auxiliary program which provides a detailed printout of all input data and a detailed description of each mine layed and its parameters as determined in program MINDEL. In addition it compiles the statistics of the number of mines layed, buried, not buried, effective, and compromised. The distributions of the mine positions are stored on magnetic tape for use in program LSQRPLOT. A detailed presentation of program MINDEL1 appears in Appendix XI. A discussion of the statistical routines appears in Section Four.

Program LSQRPLOT is an auxiliary program which computes a least squares fit to the cumulative distributions of the mine positions along the minefield X and Y axes, and presents graphs of both frequency and cumulative distributions on the axes. In addition, a random minefield is generated utilizing the statistics compiled in program MINDEL1. This program is discussed in Section Four and a program listing appears in Appendix XII.

The segmenting of the model into five programs was necessitated by the storage capacity of the CDC 1604 Computer available; however, each program is a logical portion of the model for the purposes of parametric analysis and for controlling the degree of information which may be desired by the user. In a typical employment of the model programs GRID and SURFIT would be run once to convert the geographical and bottom data for a specific mining plan. The taped data from these two programs is then available for repeated use by program MINDEL. Program MINDEL is then run for the desired number of iterations and/or for parametric anal-

ysis which may be performed by varying the punched card input data. Program MINDEL1, utilizing the taped output of program MINDEL, is then run once for each set of punched card input to program MINDEL. Following each run of program MINDEL1, program LSQRPLOT is run if the extent of the desired information includes the mine position distributions on the two axes.

The approximate storage capacities and running times for the five programs above are:

| PROGRAM | STORAGE CAPACITY | RUNNING TIME (Minutes) |
|---|---|---|
| GRID | 9,000 | 2.0 |
| SURFIT | 12,000 | 3.5 |
| MINDEL | | |
| a. One iteration | 14,000 | 5.5 |
| b. 80 iterations | 14,000 | 42.0 |
| MINDEL1 | | |
| a. One iteration | 13,500 | 3.0 |
| b. 80 iterations | 13,500 | 22.0 |
| LSQRPLOT | 24,000 | 2.5 |

This data is based on the sample run, a portion of which appears in Appendix XIII.

## 3. MINDEL: The Mine Delivery Simulation.

This section is a detailed dissertation on Program MINDEL and the background methodology, assumptions, and concepts resulting from the authors' analysis of the mine delivery operation. The real world characteristics of mining vehicles and mines involved in a mining operation have been examined and the critical aspects have been extrapolated to form a usable model as described herein.

In Program MINDEL each vehicle is considered to have arrived at the minefield objective area. The vehicles are selected in order of their arrival and each vehicle is then considered to follow a track attempting to lay its mines in their intended positions. As the vehicle progresses along its track, the attrition and damage routines appropriate to the vehicle type are considered. At each mine release point a sequence of events which is determined by the vehicle and mine characteristics is begun. The outcomes of the events in this sequence determine the final position of the mine and its parameters. With the exception of a few deterministic events, the outcomes of events are probabilistic and each outcome is resolved by comparing a random number from a uniform distribution on the interval zero to one to a probability provided by the user. If the vehicle survives and completes the transit of its assigned track, it is assumed to depart the minefield objective area and the next vehicle is selected. Figure 2 shows the basic logic of the Mine Delivery Model.

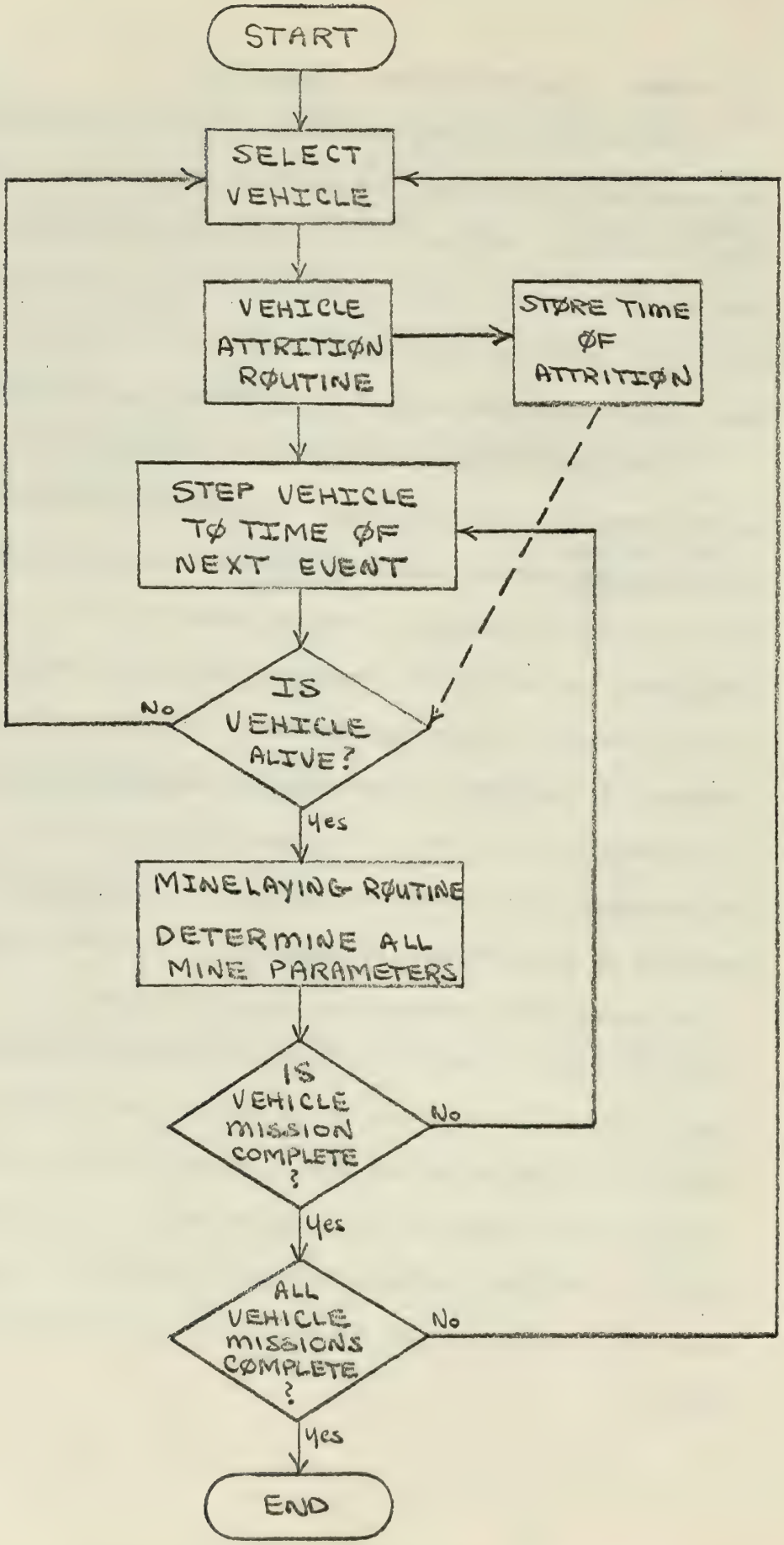



Figure 2. Flowchart of Basic Logic in the Mine Delivery Model.

A more comprehensive commentary of the authors' approach to the mine delivery simulation follows in succeeding paragraphs.

Program MINDEL uses input data from both magnetic tape and punched cards. The taped input includes:

1. Co-ordinates of the minefield perimeter.
2. Co-ordinates of the minefield reference point.
3. Vehicle leg description lists.
4. Parameters of the bottom contour and bottom type polynomials.

The punched card data includes:

1. Vehicle Arrival List.
2. Vehicle characteristics lists.
3. Mine characteristics lists.
4. Mine Description List.
5. Aircraft Bombrack Array.
6. Submarine end of mission time.
7. Array of initializing values for the pseudo-random number generator.

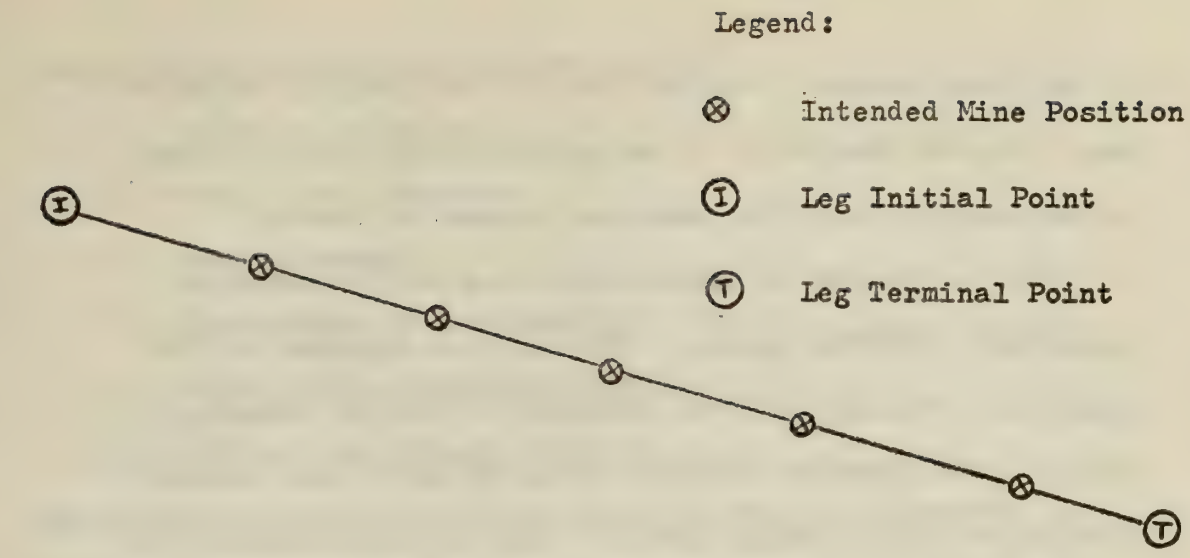
The above data is read in under indices established by the user. The indices and the array dimensions shown in the sample program listing included in Appendix IV were chosen on the basis of computer storage limitations. If a greater storage capacity is available to the user, a simple change to the appropriate dimension statements, following the guidelines of Appendix IV, will suffice to increase the force limits as desired.

The underlying methodology of the model is based on the concept of the vehicle leg. The leg is a series of points consisting

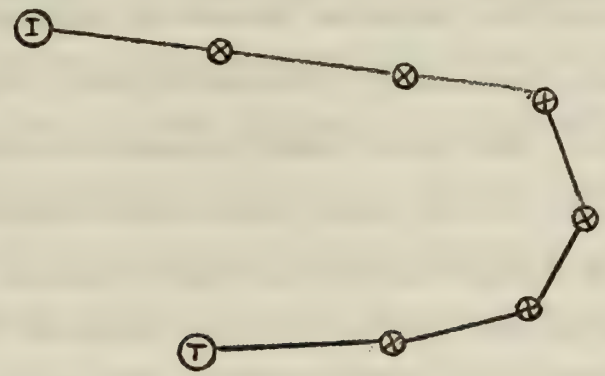
of an initial point, one or more mine intended positions (or mine firing positions in the case of submarine self-propelled mines), and a terminal point. While there are no restrictions on the orientation of a leg or on the orientation of the points on a leg with respect to each other, each pair of points in sequence is considered connected by a straight line segment (see Figure 3). If a vehicle is assigned more than one leg, the legs are considered joined by a straight line from the terminal point of each leg to the initial point of the next leg.

A vehicle, starting from the leg initial point, transits the leg by "passing through" the successive mine intended positions on the leg until it arrives at the leg terminal point. Germane to the concept of "passing through" a mine intended position are the assumptions that a vehicle always attempts to lay each mine at the intended position of that mine, and that at the time of each lay the vehicle considers itself to be at its calculated release point within the limits of its navigational capability.

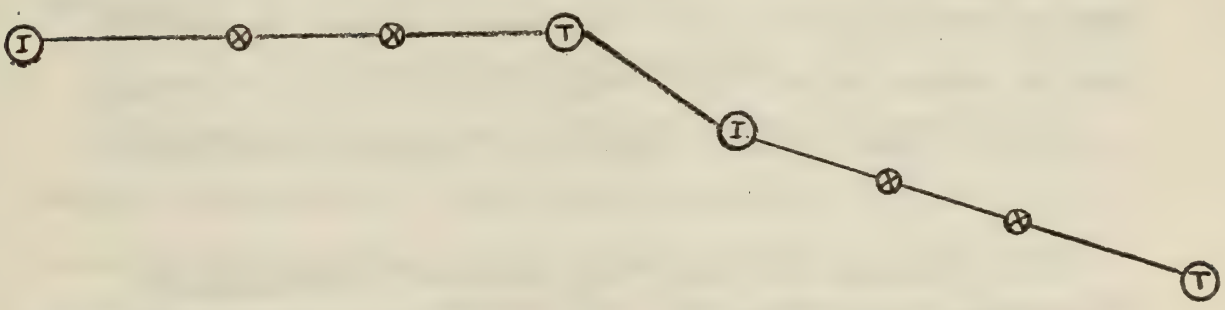
Prior to the beginning of any iterations of MINDEL a submarine end of mission time (TAM) is determined. TAM is the date-time by which submarine operations in the minefield objective area are to terminate. The user has the option of assigning a desired date-time to TAM or assigning it a zero value except when submarines are the only vehicle type used a specific value of TAM must be assigned. When a zero value for TAM is used the submarine(s) must be the first vehicle type(s) to enter the minefield objective area, and TAM is determined within the program to be the time of arrival




Example 1. Single Straight Leg




Example 2. Single Curved Leg




Example 3. Multiple Legs

Figure 3. Graphical Examples of the Leg Concept.

of the first vehicle of another type at the minefield objective area. Further discussion of the application of TAM is included in the description of the Submarine Sequence of Events.

The user assigns an appropriate value for the number of iterations of MINDEL desired. This value is dependent on the sample size the user feels is necessary to provide adequate results. Each iteration is played using the basic input data and a different set of random numbers. This is accomplished by the selection of a new initializing value for the random number generator at the beginning of each iteration. These initializing values may be selected from the set of odd integers from 1 to 67108863 inclusive and they determine the set of uniformly distributed random numbers generated by Subroutine RAN1[1] on the real interval zero to one. This random number generator is utilized throughout the program, and all other random number distributions are obtained from it through appropriate transformations.

In the discussion which follows only one iteration of MINDEL is being described.

Following the duplication of the Aircraft Bombrack Array into a temporary array, necessitated by the destructive features of the bombrack search routine in each iteration, a vehicle is selected from the Vehicle Arrival List. The Vehicle Arrival List contains, in order of their arrival at their first assigned initial points,

[1]M.C. Pike and I.D. Hill, "Algorithm 266," Communications of the ACM, VIII(October, 1965), pp. 605-606.

the following information for each vehicle:

1. Date-time of arrival.
2. Vehicle type indicator.
3. Vehicle number.
4. Total number of legs assigned.
5. Number of bombracks for aircraft.
6. Ordered list of legs assigned.

The vehicle type indicator of the selected vehicle is examined to determine which of the three basic vehicle types it is. If the vehicle type indicator is between 000 and 009 inclusive, the vehicle is a submarine type and is treated as set forth in the Submarine Sequence of Events. If the vehicle type indicator is between 010 and 099 inclusive, the vehicle is a ship type and is treated as set forth in the Ship Sequence of Events. If the vehicle type indicator is between 100 and 999 inclusive, the vehicle is an aircraft type and is treated as set forth in the Aircraft Sequence of Events. Although certain portions of the programing are common to the three vehicle sequences, the vehicle sequences of events which follow are presented separately for purposes of clarity and the common features are pointed out. The three vehicle sequences are described as follows:

## Aircraft Sequence of Events

The aircraft type number is found by subtracting 99 from the vehicle type indicator. The aircraft type number then serves as an index for entry into the Aircraft Characteristics List which contains the following information concerning each aircraft type:

of the first vehicle of another type at the minefield objective area. Further discussion of the application of TAM is included in the description of the Submarine Sequence of Events.

The user assigns an appropriate value for the number of iterations of MINDEL desired. This value is dependent on the sample size the user feels is necessary to provide adequate results. Each iteration is played using the basic input data and a different set of random numbers. This is accomplished by the selection of a new initializing value for the random number generator at the beginning of each iteration. These initializing values may be selected from the set of odd integers from 1 to 67108863 inclusive and they determine the set of uniformly distributed random numbers generated by Subroutine RAN1[1] on the real interval zero to one. This random number generator is utilized throughout the program, and all other random number distributions are obtained from it through appropriate transformations.

In the discussion which follows only one iteration of MINDEL is being described.

Following the duplication of the Aircraft Bombrack Array into a temporary array, necessitated by the destructive features of the bombrack search routine in each iteration, a vehicle is selected from the Vehicle Arrival List. The Vehicle Arrival List contains, in order of their arrival at their first assigned initial points,

[1]M.C. Pike and I.D. Hill, "Algorithm 266," Communications of the ACM, VIII(October, 1965), pp. 605-606.

the following information for each vehicle:

1. Date-time of arrival.
2. Vehicle type indicator.
3. Vehicle number.
4. Total number of legs assigned.
5. Number of bombracks for aircraft.
6. Ordered list of legs assigned.

The vehicle type indicator of the selected vehicle is examined to determine which of the three basic vehicle types it is. If the vehicle type indicator is between 000 and 009 inclusive, the vehicle is a submarine type and is treated as set forth in the Submarine Sequence of Events. If the vehicle type indicator is between 010 and 099 inclusive, the vehicle is a ship type and is treated as set forth in the Ship Sequence of Events. If the vehicle type indicator is between 100 and 999 inclusive, the vehicle is an aircraft type and is treated as set forth in the Aircraft Sequence of Events. Although certain portions of the programing are common to the three vehicle sequences, the vehicle sequences of events which follow are presented separately for purposes of clarity and the common features are pointed out. The three vehicle sequences are described as follows:

## Aircraft Sequence of Events

The aircraft type number is found by subtracting 99 from the vehicle type indicator. The aircraft type number then serves as an index for entry into the Aircraft Characteristics List which contains the following information concerning each aircraft type:

1. Aircraft type indicator.
2. Transit speed (knots).
3. Minelaying speed (knots).
4. Minelaying altitude (feet).
5. Probability of mine release.
6. Probability of survival over the objective area.

The minelaying speed is converted to yards per minute and a vehicle type flag is set to control the flow of this sequence in subsequent portions of the program.

The total number of legs and the number of the first leg assigned to the aircraft are extracted from the Vehicle Arrival List. Utilizing the leg number the corresponding leg data is extracted from the Aircraft Leg Description List which contains the following information for each aircraft leg:

1. Leg number.
2. Number of mines to be layed on the leg.
3. X and Y co-ordinates of the leg initial point.
4. For each mine on the leg, the mine number and the X and Y co-ordinates of its intended position.
5. X and Y co-ordinates of the leg terminal point.

The distance between consecutive points on the leg and the time increments to transit those distances at minelaying speed are computed. The time increments are summed and represent an ordered record of the time elapsed (at all points on the leg) relative to the vehicle's time of arrival at its first initial point. If the vehicle is assigned more than one leg, the time increment for the distance between the terminal point and the initial point of the

next leg is computed and stored in the ordered record of relative times. All succeeding legs assigned to the vehicle are handled in an identical manner.

Upon completion of the calculation of the record of relative times, it is next determined whether the aircraft is attrited and, if so, at what time? A uniform random number is compared to the aircraft probability of survival for the aircraft type. If the aircraft is attrited, the time of attrition is found by multiplying the aircraft's total time in the minefield objective area by a uniform random number. This time is referred to as "dead time".

With completion of the computation of the foregoing values, the actual simulation of the aircraft mining sequence is begun. The aircraft transits to its first assigned mine release point and the relative time for arrival at this point is compared to dead time. If dead time is the earlier of the two times, the aircraft is attrited and "true dead time" is calculated by adding dead time to the date-time of arrival of the aircraft at its first initial point. The attrition event along with the numbers of the mines remaining on board the aircraft are recorded and the next vehicle is selected. If the aircraft is not attrited at this time, the following sequence of events is initiated:

1. Mine release from aircraft.
2. Parachute deployment.
3. Water impact.
4. Actual position determination.
5. Bottom impact.

6. Burying (bottom mines only).

7. Case and anchor separation (moored mines only).

8. Arming (moored mines only).

9. Cable deployment (moored mines only).

10. Reliability.

The mine release event is begun with a search of the Aircraft Bombrack Array. The array contains a list of the mines, by mine number in order of intended release, as distributed among the six possible bombracks for each aircraft. The number of each mine in a releaseable position is compared with the number of the mine to be layed at this position, as obtained from the Aircraft Leg Description List. If the search is unsuccessful, the mine is considered to be blocked by a prior release failure. If successful, a uniform random number is compared with the probability of release for the aircraft. If the mine does not release or is blocked, this information is recorded and the aircraft proceeds to the next point on the leg. If the mine releases, that mine number is deleted from the bombrack array and the next mine moves into a releaseable position. The information that a mine has been released along with the true time of release is recorded and the aircraft proceeds to the next point on the leg. This sequence of events is repeated until the aircraft is either attrited or arrives at its leg terminal point. If the aircraft has been assigned more than one leg, and it arrives at its next leg initial point without being attrited, the above sequence is repeated for all succeeding legs.

The Aircraft Mine Characteristics List contains the following information:

1. Mine type (II).
2. Probability of parachute deployment.
3. CEP for 3K feet, given that the chute deploys.
4. CEP for 3K feet, given that the chute does not deploy.
5. Probability of water impact damage, given that the chute deploys.
6. Probability of water impact damage, given that the chute does not deploy.
7. Probability of impact damage on the hardest bottom, given that the water depth is less than 20 feet and the chute deploys.
8. Probability of impact damage on the hardest bottom, given that the water depth is less than 20 feet and the chute does not deploy.
9. Probability of burying in the softest bottom.
10. Indicator of moor or bottom type mine.
11. NWL type if bottom mine does not bury/buries.
12. Probability of case/anchor separation.
13. Probability of full anchor cable deployment.
14. Probability that the mine operates, given that it has not been damaged (reliability).

When a mine is released by an aircraft, a uniform random number is compared to the probability of parachute deployment. If the parachute deploys, or the mine is not parachute equipped, the corresponding CEP for an aircraft altitude of 3000 feet, the probability of water impact damage, and the probability of impact damage on the hardest bottom given that the water depth is less than 20 feet

are extracted from this list. If the parachute does not deploy, the corresponding probabilities are extracted in place of those mentioned above. In either case, an appropriate modification is made to the probability of burying in the softest bottom when the depth of water at the impact point is found to be less than 20 feet. A uniform random number is then compared to the probability of damage on water impact. If damage occurs, this information is recorded.

The mine's corrected CEP is a function of the appropriate original CEP for an aircraft altitude of 3000 feet and the aircraft's minelaying altitude (see Appendix VI for mathematical background). The corrected CEP is converted to a standard deviation which is used as an input to Subroutine RNORM. This subroutine generates the X and Y errors which are applied to the intended position to get the final position of the mine.

Using the final position of the mine determined above, Subroutine DEPTH is called to find the water depth at that point. If the calculated water depth is negative, the fact that the mine has been layed on the beach is recorded and the vehicle proceeds to the next point on its leg. If the water depth is positive, Subroutine DEPTH is then called again; this time with the necessary arguments for determining the bottom type value at the mine's final position.

The Mine Description List contains the following information about each mine:

1. Mine number.

2. Mine type (II).

3. Mine type (I).

4. Mine case depth (intended).

5. Arming delay.

6. Sterile time.

7. Ship counts.

8. Type of moor.

Depending on whether the mine layed is a moored mine or a bottom mine, as determined from the above list, the sequence of events now splits into two paths. For moored mines, the water depth at the mine's final position is tested. If the water depth is less than 20 feet and the mine's parachute failed to open, the appropriate probability of damage on the hardest bottom is used in conjunction with the calculated bottom type factor to determine the probability of damage on bottom impact. If the mine's parachute did deploy, the same calculations are made using the alternate value of the probability of damage on the hardest bottom. A uniform random number is then compared to the probability of damage on bottom impact. When damage is found to have occurred, this fact is recorded. If the water depth at the mine's final position is found to be greater than 20 feet, the probability of damage on bottom impact is considered to be negligible.

To determine whether case and anchor separate, a uniform random number is compared to the probability of case/anchor separation. If separation does not occur this event is recorded and if the mine requires case/anchor separation in order to arm,

the failure to arm is also recorded (the fact that the mine does or does not require case/anchor separation in order to arm is determined from the indicator of moor or bottom type in the Aircraft Mine Characteristics List). For moored mines not requiring case/anchor separation to arm, reliability is tested as below. The mine is then considered to come to rest on the bottom at its final position and the aircraft proceeds to its next point. If the case and anchor separation takes place, the case depth is then determined by comparing a uniform random number to the probability of full cable deployment. If the mooring cable fails to deploy fully, the intended full cable deployment length is multiplied by a uniform random number; otherwise the cable deploys fully in accordance with its case depth setting. In both cases the final case depth is calculated by two techniques, one for fixed case depth moored mines and another for fixed cable length moored mines; both of which utilize intended case depth and water depth at the mine's final position. Finally, given that the mine has not been damaged, it is determined and recorded whether the mine operates or not by comparing a uniform random number and the mine's reliability.

For bottom mines, the initial steps involving the determination of the probability of damage on the bottom are the same as for moored mines. In addition, the probability of burying, as a function of the probability of burying in the softest bottom and bottom type, is determined. This probability is modified by the addition of a factor of .1 if the water depth is less than 20

feet and the mine's parachute deployed or by the addition of a factor of .2 if the water depth is less than 20 feet and the mine's parachute did not deploy. It is determined if the bottom mine is a type for which burying is desirable or not desirable (by checking the indicator of moor or bottom type for the mine in the Aircraft Mine Characteristics List) and a uniform random number is then compared to the probability of burying. If the mine is designed to bury and it does not or if it is designed not to bury and it does, a reduced actuation sensitivity (NWL type if bottom mine does not bury/buries in the Aircraft Mine Characteristics List) is selected and the event is recorded. Finally, if the mine has not been damaged, a uniform random number is compared to the mine reliability and the outcome is recorded.

Following the laying of a bottom or moored mine the events as recorded are written on magnetic tape and the aircraft proceeds to its next point.

It should be noted at this point that both "damaged" and "unreliable" here carry a special connotation. "Damaged" implies that the mine has been rendered inoperable. "Unreliable" implies that although the mine is not damaged in the laying sequence, it fails, totally, to operate.

### Ship Sequence of Events

The ship type number is found by subtracting nine from the vehicle type indicator. The ship type number then serves as an index for entry into the Ship Characteristics List which contains the following information for each ship type:

1. Ship type indicator.

2. Transit speed (knots).

3. Minelaying speed (knots).

4. Navigational error (yards).
   a. Least.
   b. Average.
   c. Greatest.

5. Probability of mine release.

6. Maximum time delay.

7. Probability of some damage to the ship type for the first half-hour in the minelaying objective area.

The minelaying speed is converted to yards per minute and a vehicle type flag is set to control the flow of this sequence in subsequent portions of the program. Additionally, the navigational error values are extracted from the Ships Characteristics List and a flag indicating that no ship damage has occurred is set.

Ship leg information and the record of relative times is determined from the Ship Leg Description List, which is identical in format to the Aircraft Leg Description List. Upon completion of the foregoing and utilizing the probability of some damage for the first half-hour in the minelaying objective area and the total time in the minefield, the probability of some damage occurring is determined and tested as shown in Appendix VIII. If damage does not occur, the time of damage is set to a value greater than the ship's total time in the field. If the ship does suffer some damage, the time of damage is determined by multiplying the ship's total time in the field by a uniform random number. This time is again referred to as "dead time".

The actual simulation of the ship mining sequence is now begun. The ship transits to its first mine release point and the probability of release is compared to a uniform random number to determine if the mine releases on the first attempt. Ship layed mines are considered to always release due to the simplicity of ship mine racks and the availability of alternative means of launching the mine. However, if the mine does not release on the first attempt, a time delay is computed by multiplying the maximum time delay by a uniform random number. The X and Y distances traveled during this delay time are calculated for use in the navigational error routine and the time delay is added to the intended time of release. The computed release time is then compared to "dead time". If "dead time" is the earlier of the two times, the ship is considered to have suffered some damage at "dead time", and the event is recorded. The type of damage is randomly determined from among the following:

1. Speed reduced by one-half.
2. Navigational error doubled.
3. Sunk.
4. Aborts mission.
5. No effect on mission.

The applications of the type of damage are discussed in Appendix VIII, and only consider damage in the ways it may affect the minelaying mission. Following the determination of the type of damage, appropriate ship parameters are adjusted or the mission is terminated. When a ship minelaying mission is terminated,

this event and the mines remaining on board the ship are recorded and the next vehicle is selected.

If the mission continues, the damage flag is set so no further tests for damage will be made and when "dead time" is later than the computed release time the mine is released at the computed release time.

When a mine is released the following sequence of events is initiated:

1. Actual position determination.
2. Bottom impact.
3. Burying (bottom mines only).
4. Case and anchor separation (moored mines only).
5. Arming (moored mines only).
6. Cable deployment (moored mines only).
7. Reliability.

The basic logic for determining the outcomes of these events is essentially the same as that described in the Aircraft Sequence of Events. The only significant exception being the navigational error routine, which is described in Appendix VII. The various probabilities and indicators in this instance are taken from the Ship Mine Characteristics List which contains the following information for each ship layed mine type:

1. Mine type (II).
2. Probability of damage on the hardest bottom, given that the depth is less than 20 feet.
3. Probability of burying in the softest bottom.
4. Indicator of moor or bottom type.

5. NWL type if bottom mine buries/does not bury.

6. Probability of case/anchor separation.

7. Probability of full anchor cable deployment.

8. Reliability.

The ship continues to transit each of its assigned legs until it arrives at the terminal point of its last leg or is sunk or aborts the mission. All the events are recorded and the outcomes of the laying events are written on magnetic tape.

Submarine Sequence of Events

The submarine type number is set equal to the vehicle type indicator. The submarine type number then serves as an index for entry into the Submarine Characteristics List which contains the following information concerning each submarine type:

1. Submarine type indicator.

2. Transit speed (knots).

3. Minelaying speed (knots).

4. Navigational error (yards).
   a. Least
   b. Average
   c. Greatest

5. Probability that the mine is ready at firing time (PMR).

6. Maximum firing time delay (FTDM) (minutes).

7. Probability of detection (PD).

8. Probability that the submarine has knowledge of a detection, given that a detection has occurred (PKD/D).

9. Probability that the submarine thinks detection has occurred, given no detection (PKD/ND).

10. Probability that the submarine suspends operations, given it has knowledge of detection (PSO/KD).

11. Probability that the submarine will return, given it has suspended operations (PR/SO).

12. Probability that the submarine will return, given an unsuccessful attack (PR/UA).

13. Probability of attack, given a detection (PA/D).

14. Probability of a successful attack, given an attack (PSA/A).

15. Distance from the minefield perimeter to the one-hundred fathom curve (D(100)).

The minelaying speed is converted to yards per minute and a vehicle type flag is set to control the flow of this sequence in subsequent portions of the program. Additionally, the time required for the submarine to reach the 100 fathom curve from the minefield using transit speed is calculated and the navigational error values are extracted.

Submarine leg information and the record of relative times is determined from the Submarine Leg Description List in basically the same manner as was described for aircraft. The Submarine Leg Description List contains the following information for each submarine leg:

1. Leg number.

2. Number of mines to be layed on the leg.

3. X and Y co-ordinates of the leg initial point.

4. For each mine on the leg, the mine number, the X and Y co-ordinates of the submarine's firing position, and the X and Y co-ordinates of the mine's intended position.

5. X and Y co-ordinates of the leg terminal point.

The additional information provided by the X and Y co-ordinates of the submarine firing position is necessitated by the submarine's capability to lay self-propelled mines.

It is obvious upon examination of the submarine mining operation that the submarine "detection/attrition" routine can not be as simply modeled as that of aircraft and ships. The submarine's best defense is evasion and the fact that it conducts the mining operation covertly dictates complex techniques for simulating the numerous possible combinations of detection, attack, and evasive tactics. The several probabilities concerning detection, attack, and tactics are extracted from the Submarine Characteristics List and are combined to form a set of exhaustive and mutually exclusive events as described in Appendix IX. One of the most important tactics provided for is that the submarine can leave the field and return at a later time, but it is subject to detection and attack after each subsequent return. Following the combining of the several probabilities above, "dead time" is calculated by multiplying the submarine's expected time in the minelaying objective area by a uniform random number. In this instance "dead time" is the occurrence of one of the possible detection/attrition events.

The actual simulation of the submarine mining sequence is begun with the submarine transiting to its first firing position. The probability that the mine is ready is compared to a uniform random number. If the mine is not ready to fire, a time delay in tenths of a minute is found by generating a uniform random number. The delay is then compared to the maximum allowable time delay

(FTDM). If the time delay is less than the maximum allowable time delay, it is added to the original firing time. This new firing time is then compared to the original firing time of the next mine. If the new firing time is the earlier of the two times, the test to determine if the mine is ready is made again. The cycle is repeated until the mine is determined to be ready to fire, or the maximum allowable time delay is exceeded, or the firing time of the next mine is approached. In the latter two cases the mine is not layed, the information is recorded, and the submarine proceeds to its next firing point. This cycle of events is included to simulate the submarine's unique capability to maneuver as necessary to maintain firing position when a delay occurs. However, when the firing delay becomes excessive, the submarine will elect to leave a gap in the field due to the complexities of rearranging its firing sequence plan and mine storage.

When a mine is found ready to be fired, the original (or revised) firing time is compared to "dead time". If "dead time" is the earlier of the two times, the type of detection/attrition event is randomly determined as described in Appendix IX. Although there are fourteen possible detection/attrition events, each of which is handled differently, they can be summarized by stating that the submarine takes no action, is attrited or aborts the mission, or leaves the field and returns at some later time. If the submarine is attrited or aborts the mission, the event and the mines remaining on board are recorded, and the next vehicle is selected. When the submarine elects to leave the field, the time

out of the field is determined by the cause of the submarine leaving the field and is primarily a function of the transit time to the 100 fathom curve. Having left the field the submarine can only return if its time of return does not exceed the submarine end of mission time (TAM). Upon return the submarine resumes minelaying operations at the point of suspending operations and a new "dead time" is computed. When the submarine takes no action, or the original (or revised) firing time is earlier than "dead time", the firing time is compared to TAM and the submarine either terminates operations or the mine is fired.

When a submarine mine is fired, the following sequence of events is initiated:

1. Determination of failure for self-propelled mines.
2. Actual position determination.
3. Bottom impact.
4. Burying (bottom mines only).
5. Case and anchor separation (moored mines only).
6. Arming (moored mines only).
7. Cable deployment (moored mines only).
8. Reliability.

The basic logic for determining the outcomes of the above events is essentially the same as that described in the Aircraft Sequence of Events. The two significant exceptions are the navigational error routine, which is the same as that for ships (see Appendix VII), and the self-propelled mine position and failure routine,

which is described in Appendix X. The various probabilities and indicators in this instance are taken from the Submarine Mine Characteristics List which contains the following information for each submarine layed mine type:

1. Mine type (II).
2. Probability of damage on the hardest bottom.
3. Probability of burying in the softest bottom.
4. Indicator of moor or bottom type.
5. NWL type if bottom mine buries/does not bury.
6. Probability of case/anchor separation.
7. Probability of full anchor cable deployment.
8. Probability of a normal run for a submarine self-propelled mine.
9. Probability of a motor failure, given a failure for a self-propelled mine.
10. Probability of a gyro failure, given a failure for a self-propelled mine.
11. Probability of a depth failure, given a failure for a self-propelled mine.
12. CEP for a normal run of a self-propelled mine.
13. Reliability.

The submarine continues to transit each of its assigned legs until it arrives at the terminal point of its last leg or otherwise terminates the mission. All events are recorded and the outcomes of the laying events are written on magnetic tape.

The foregoing is an attempt to provide a description of the general logic flow in the mine delivery simulation. It should be evident that numerous devices not mentioned here, such as

indices and flags, have been used to direct the logic flow. The application of these devices is apparent from the program listing, list of variable names, and flow charts which appear in Appendices IV and V.

4. Applications of the Mine Delivery Model.

Now that the Mine Delivery Model has been described, what are its uses? In brief, the model's purpose is to examine and simulate in detail the effects of the delivery vehicles characteristics, individual mine characteristics, and bottom characteristics on the final minefield. The general technique in use today in the modeling community is the generation of a planned minefield by some selected probability distribution with the normal and uniform distributions being most popular for the purposes of threat assessment. It is the authors' contention that this approach does not adequately reflect the effects of vehicle characteristics, such as maneuverability and navigational error, on the final minefield pattern regardless of the randomness intended. Also, it ignores the effects of bottom characteristics. If it is not already apparent from the description of the simulation that there are several factors regarding each mine which will influence the effectiveness of the resultant minefield, it is hoped that the reader will be convinced of this fact as well as the limitations of assuming some distributions on the axes when he has finished reading this section.

There are three general categories of information available from the use of the Mine Delivery Model. The first is the Summary Narrative (MINDEL) which presents information pertaining to the delivery vehicles. In addition to the record of events occurring to each vehicle, this narrative indicates the mines layed or not layed, and why, by mine number. This information

is of limited applicability. Since vehicle attrition is directly related to the inputs provided by the user, vehicle attrition is of value only with respect to its effect on which particular mines are layed and the total number of mines layed. Additionally, although no vehicle interaction is programed into the model, the chronological record of events provides the user with a limited capability for investigating such interaction.

The second category of information is the Detailed Narrative (MINDEL1) which presents the results of the laying sequence for each mine layed plus the following five statistics for each iteration:

1. Number of mines layed.
2. Number of mines effective.
3. Number of mines buried.
4. Number of mines not buried.
5. Number of mines compromised.

The number of mines layed per iteration directly reflects the success of the delivery operation. This coupled with the number of effective mines and the number of mines buried or not buried determines the effectiveness of the minefield. Although the mines layed and not effective are not known to be considered in existing threat assessment models, the parameters of such mines are included in the Detailed Narrative in anticipation of the introduction of minehunting features into threat assessment models. The number of mines compromised is considered of interest to the minefield planner because of the loss of the element

of surprise for covert fields, and because of the fact that the enemy's countermeasures effectiveness is greatly enhanced by the recovery of even a single mine.

The third and last category of information is the distributions of mine positions (LSQRPLOT). The program LSQRPLOT is included in this paper as one possible course of action utilizing the frequency and the cumulative distributions generated. Samples of the graphical output of program LSQRPLOT appear in Figures 8, 9, 10, 11, 12 and 13, and the program listing appears in Appendix XII.

As pointed out in the opening paragraph of this section, the procedure for generating mine positions for threat assessment models today involves the use of one of a few standard probability distributions. It has been our purpose to demonstrate that the final result of a mine delivery operation is not a purely random field, but bears the signatures of the vehicles which layed the field. Figure 4 shows the intended positions of the 167 mines which could be representative of a planned minefield. Figures 5, 6, and 7 represent the possible results of the execution of this mining plan. It should be apparent that:

1. Vehicle navigational error plays an important role in the final minefield distribution. The distortion to the intended patterns introduced by the aircraft layed mines, based on the input data, is most evident in the eastern portion of the sample minefields.
2. The effect of vehicle attrition on the intended minefield pattern is a function of the time of attrition and the

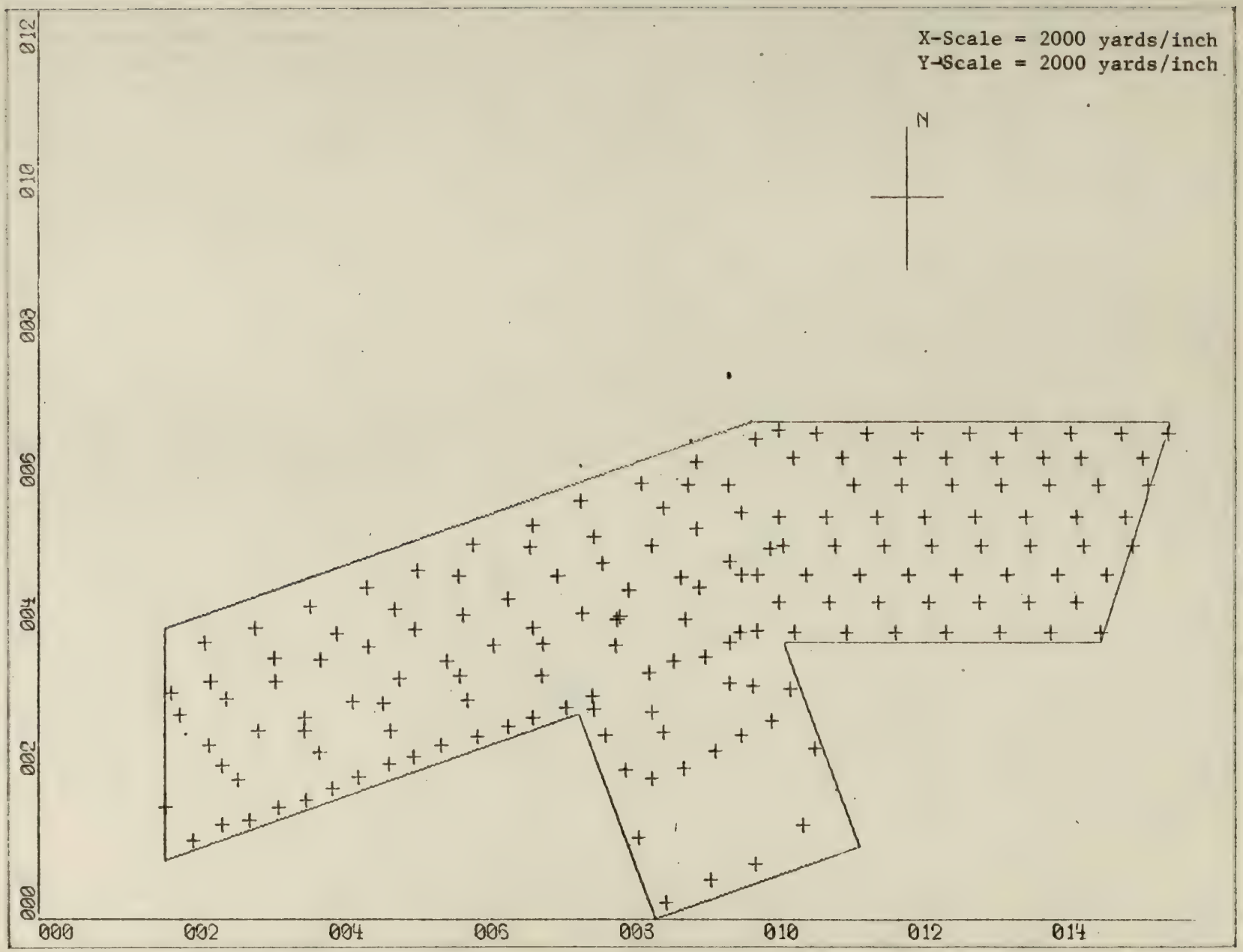



Figure 4. Intended Mine Positions.

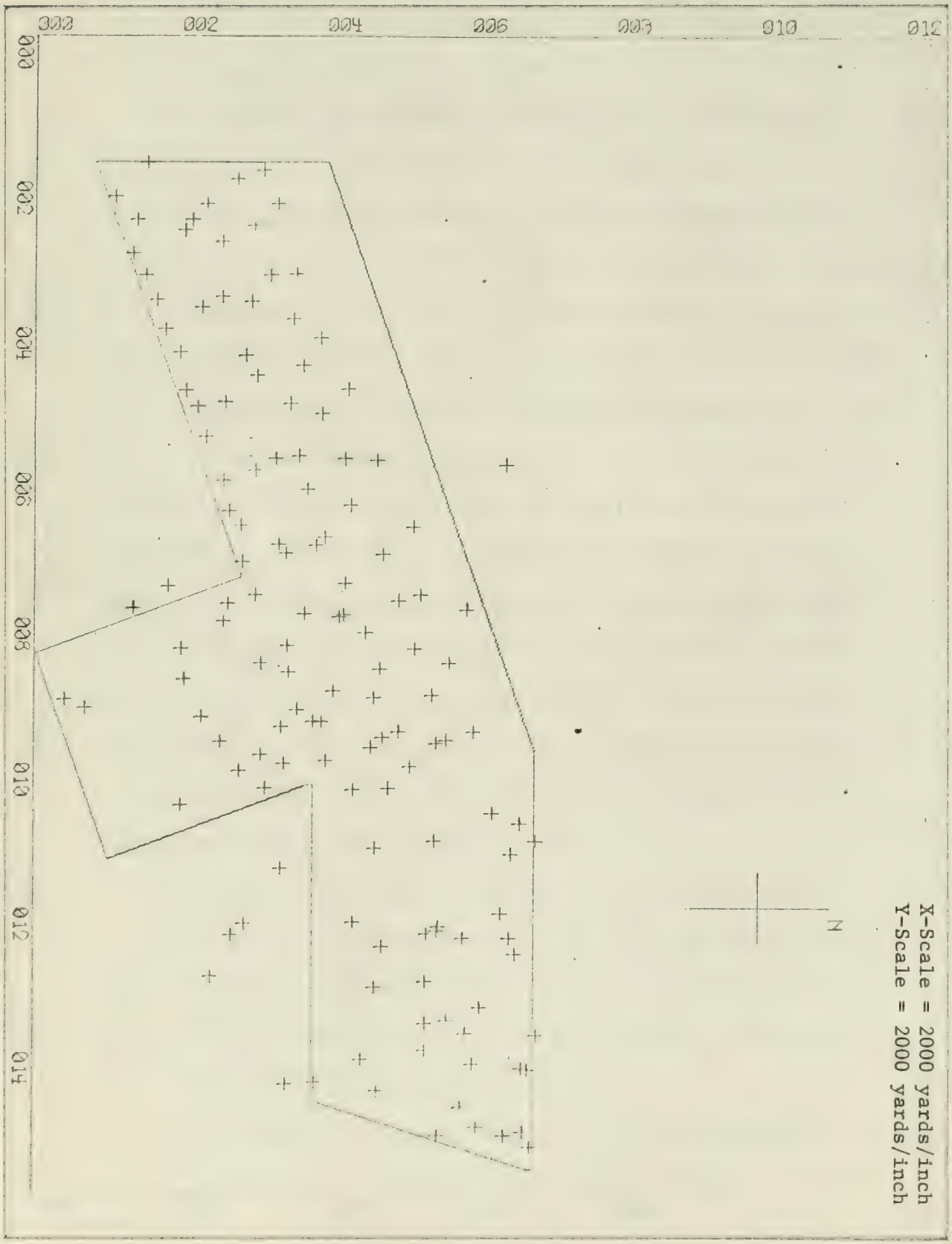



Figure 5. Sample Run - Final Mine Positions Generated by MINDEL.

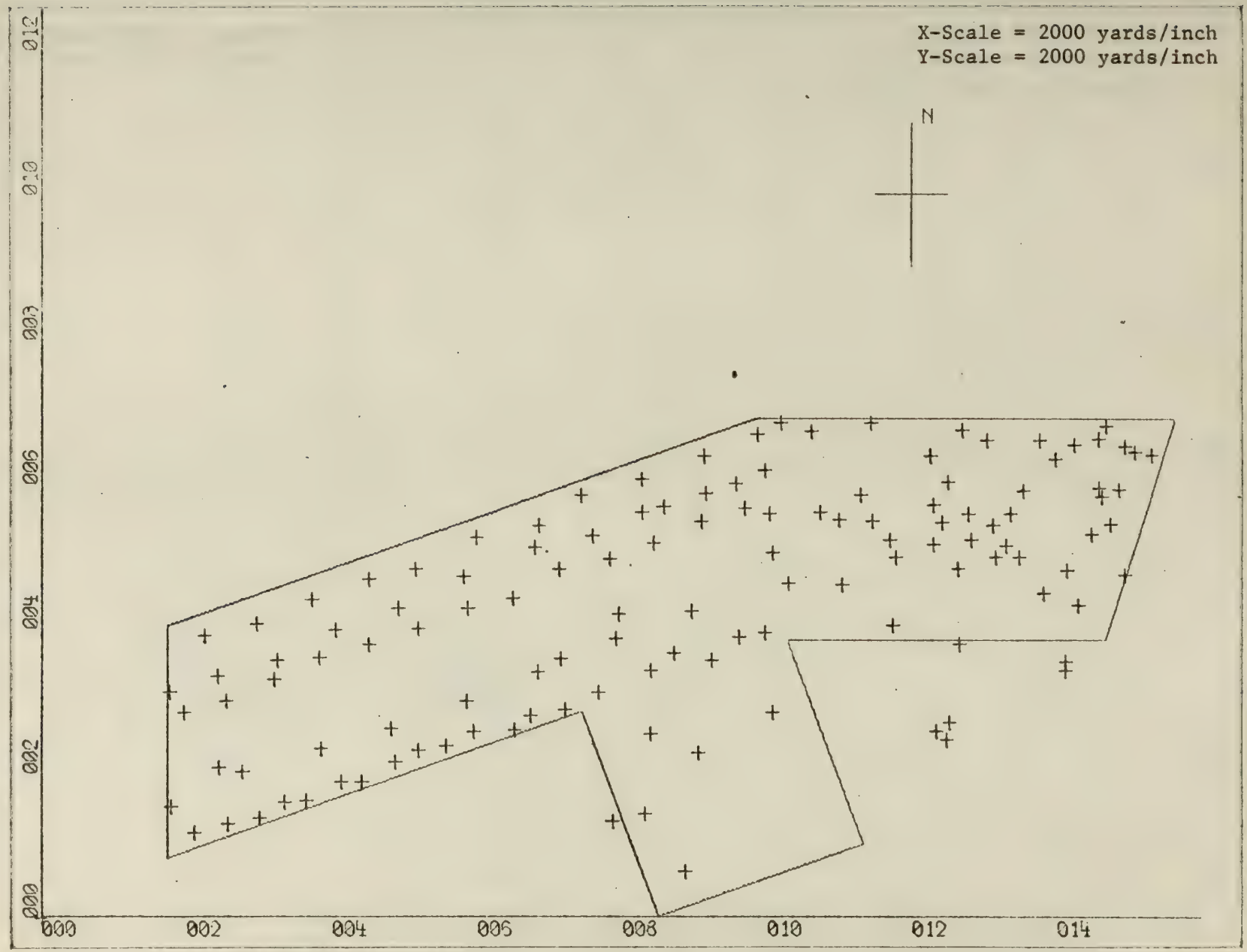



Figure 6. Sample Run - Final Mine Positions Generated by MINDEL.

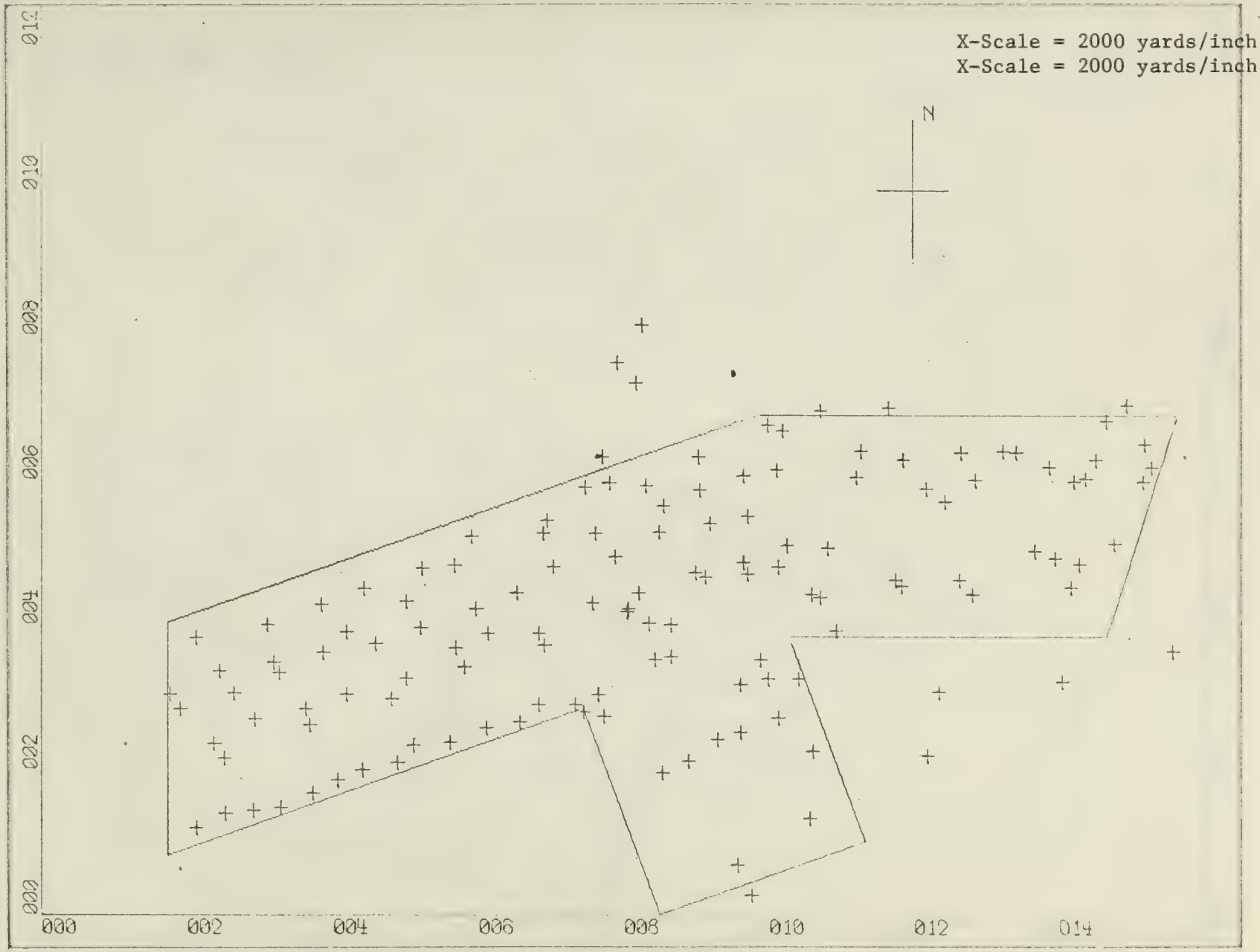



Figure 7. Sample Run - Final Mine Positions Generated by MINDEL.

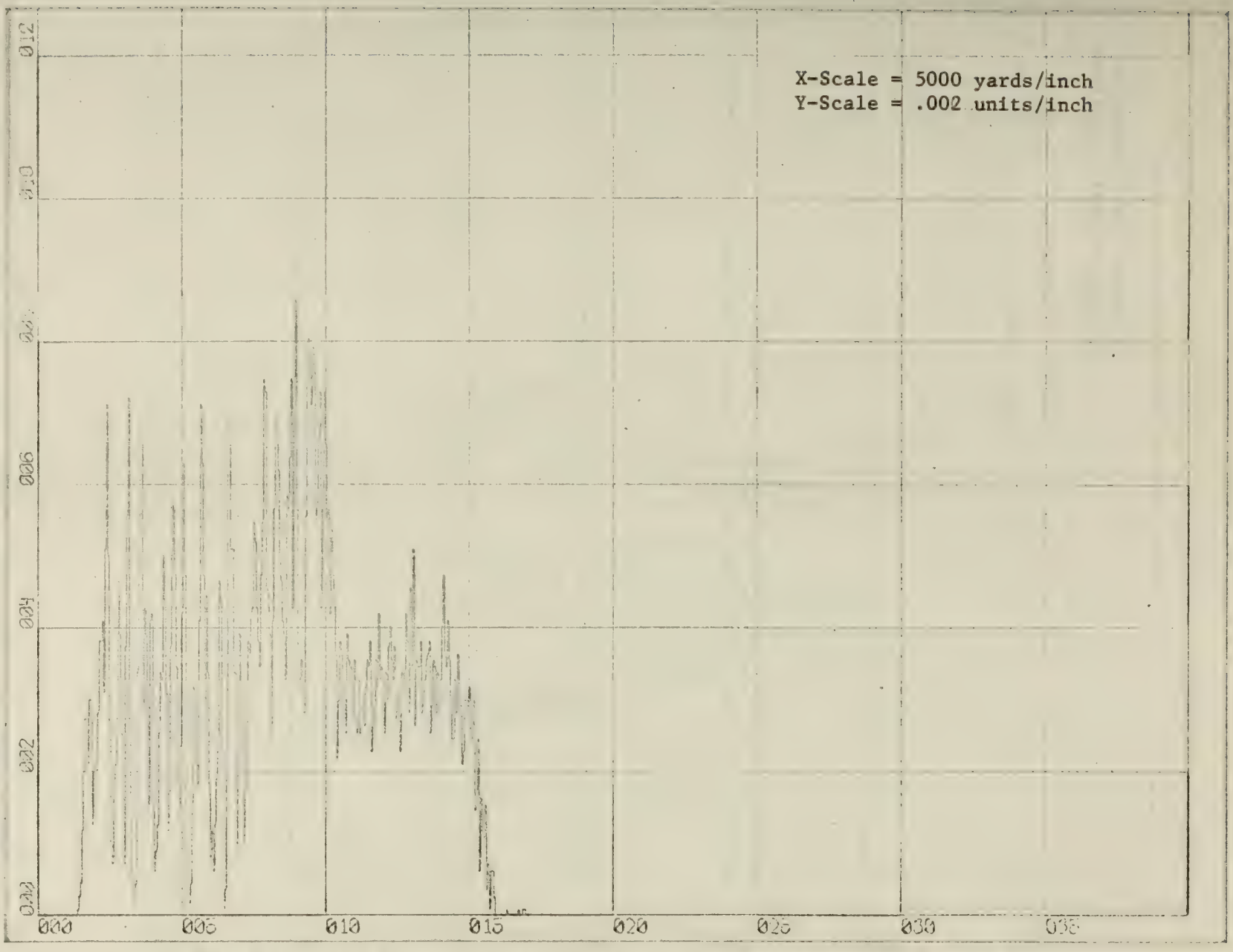



Figure 8. Mine Frequency Distribution on the X-Axis for all Sample Runs.

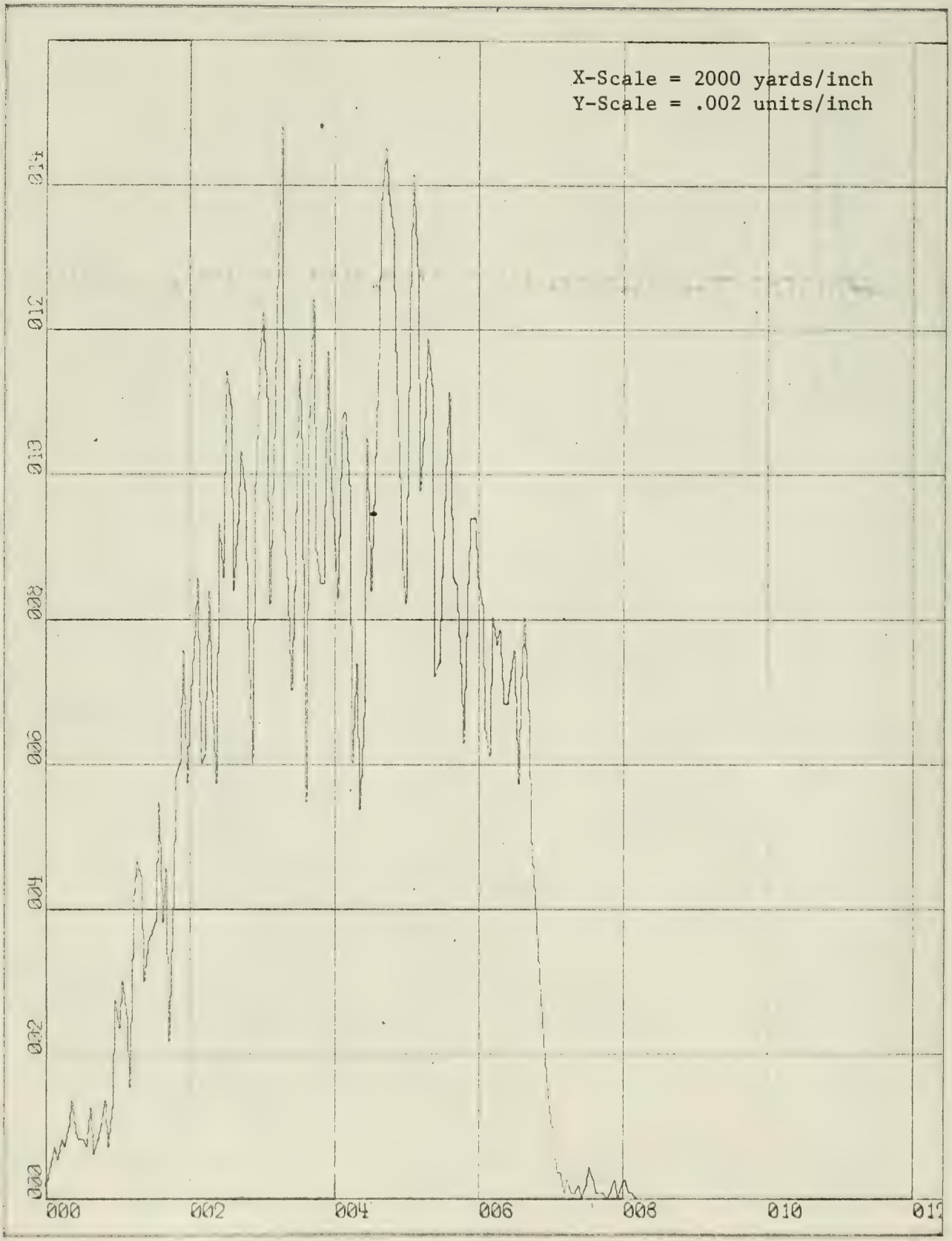



Figure 9. Mine Frequency Distribution on the Y-Axis for All Sample Runs.

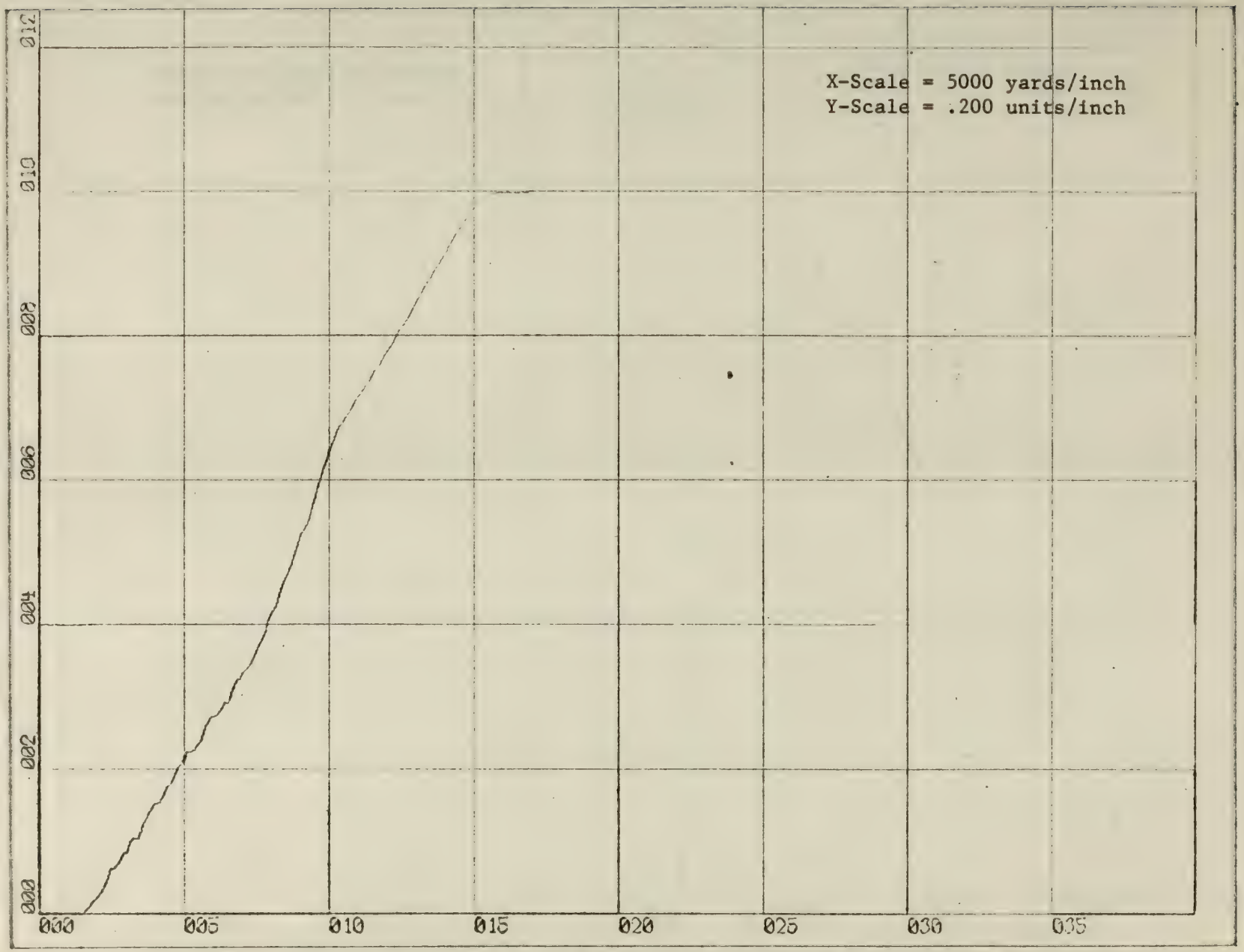



Figure 10. Mine Cumulative Distribution on the X-Axis for All Sample Runs.

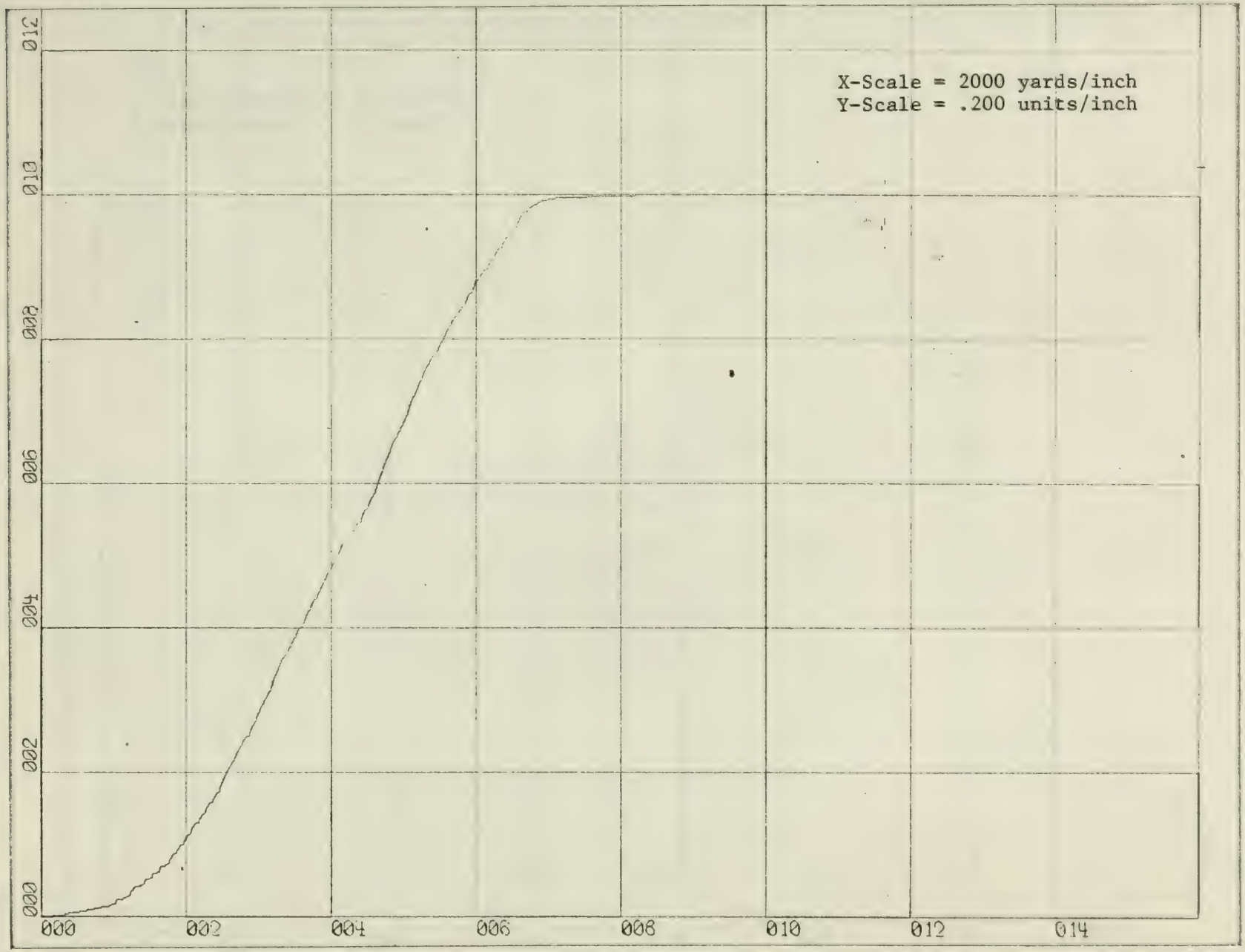



Figure 11. Mine Cumulative Distribution on the Y-Axis for All Sample Runs.

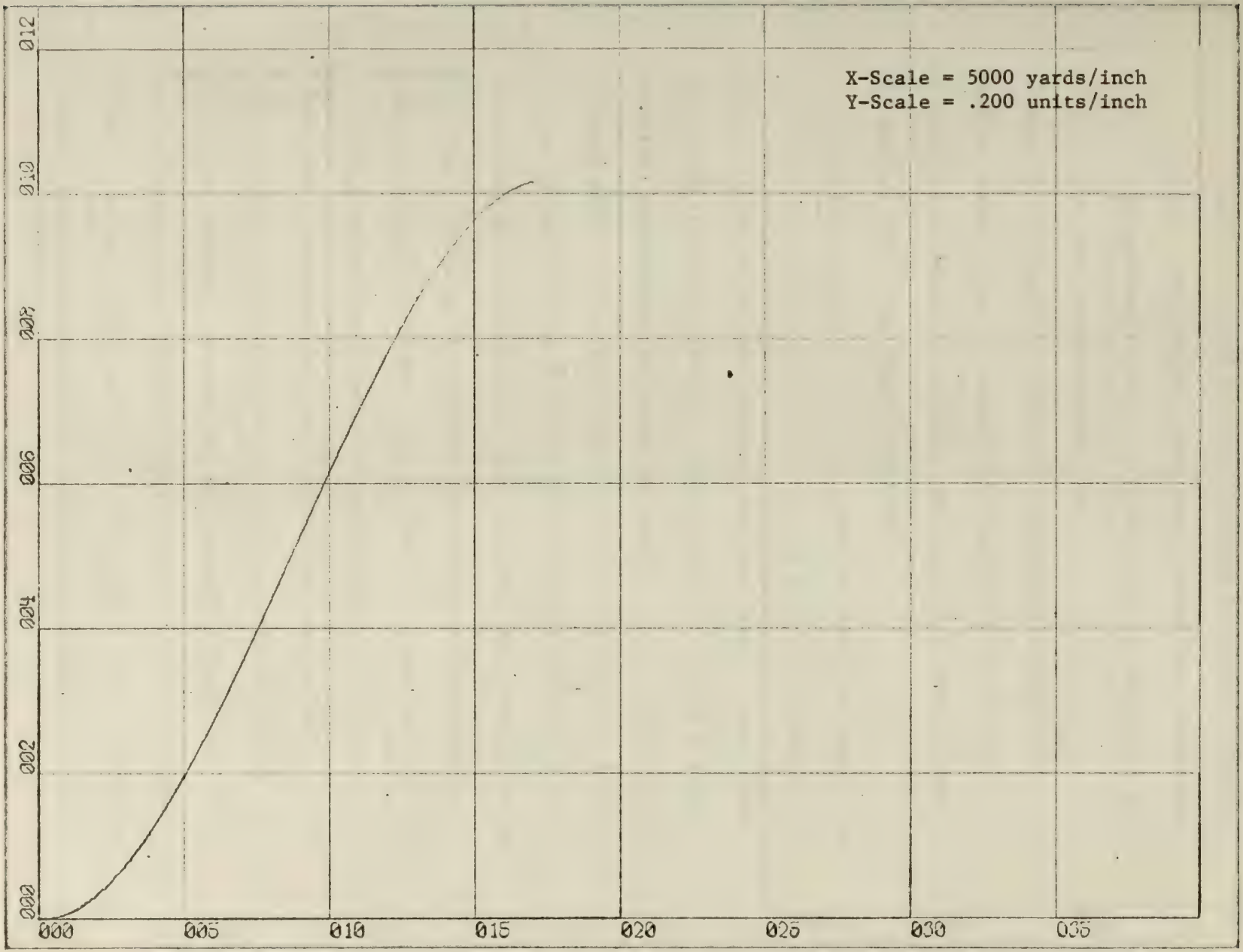



Figure 12. Least Squares Fit to Mine Cumulative Distribution Curve on the X-Axis for All Sample Runs.

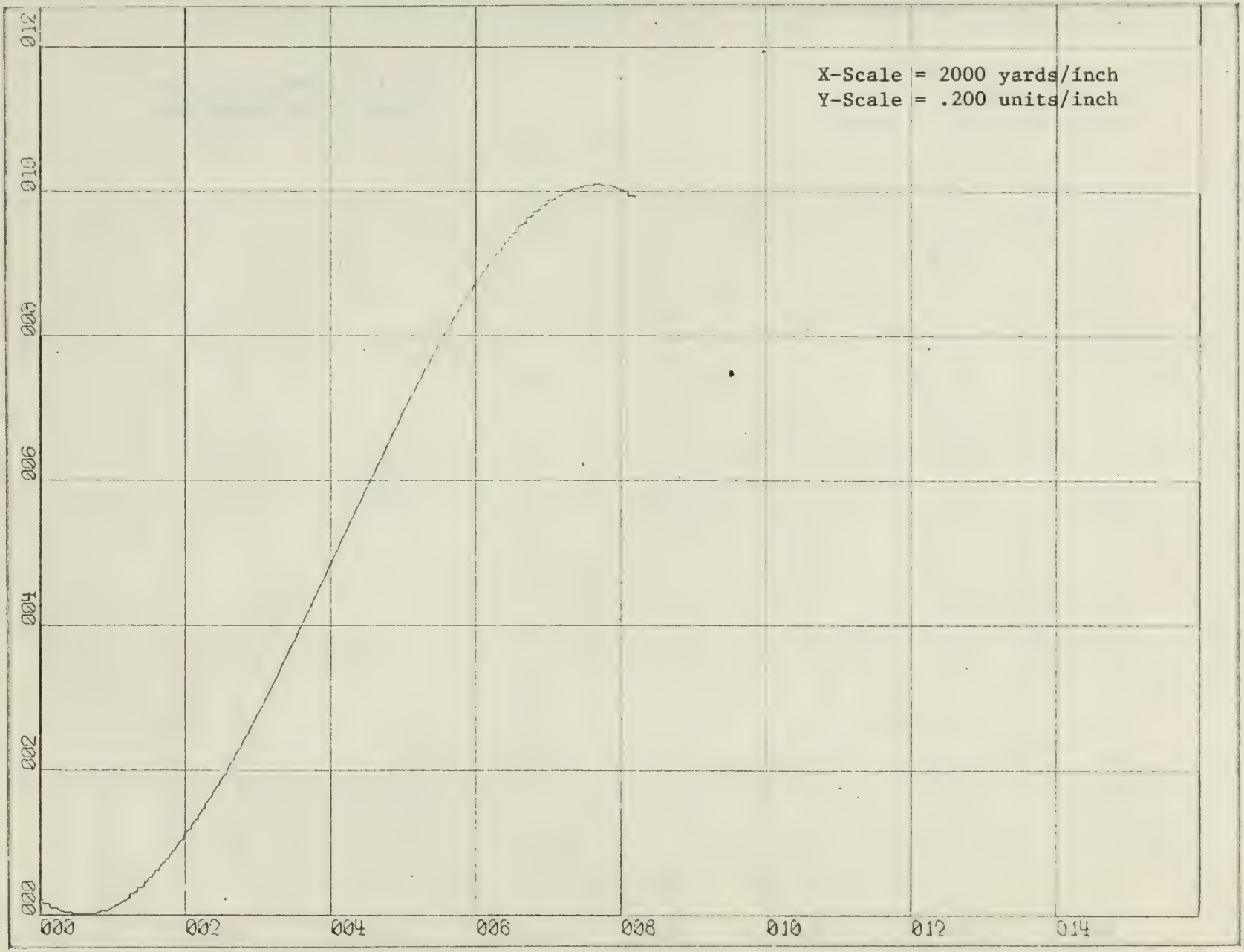



Figure 13. Least Squares Fit to Mine Cumulative Distribution Curve on Y-Axis for All Sample Runs.

vehicle mine load. This can not be simulated by random removal of mines from the intended minefield. The obvious "gaps" in the three sample minefields shown demonstrate the important effect of vehicle attrition on the final minefield pattern.

In view of the above and of the fact that consideration of vehicle capabilities is inherent to minefield planning and execution, there should be no doubt remaining that vehicle characteristics affect the final minefield.

Our primary objective of this model has been the generation of a "most probable" minefield for use by a threat assessment model. It was hoped that the data obtained from a large number of runs of this model could be analytically reduced to a single minefield representing the expected results of the execution of a mining plan. It has been suggested that this could be accomplished by defining the mine position distributions on two reference axes of the minefield. This approach was investigated and the results of using the observed distributions to generate a random minefield are shown in Figure 14. It is obvious upon comparison of the randomly generated pattern with the intended and other sample patterns that this approach is unsatisfactory. This general scheme can be improved upon, if the user is willing to expend the computer time and has the computer storage available, by developing several distributions on one axis, each of which is valid for a specified range on the other axis. Still unanswered, however, is the question of how many distributions on the one

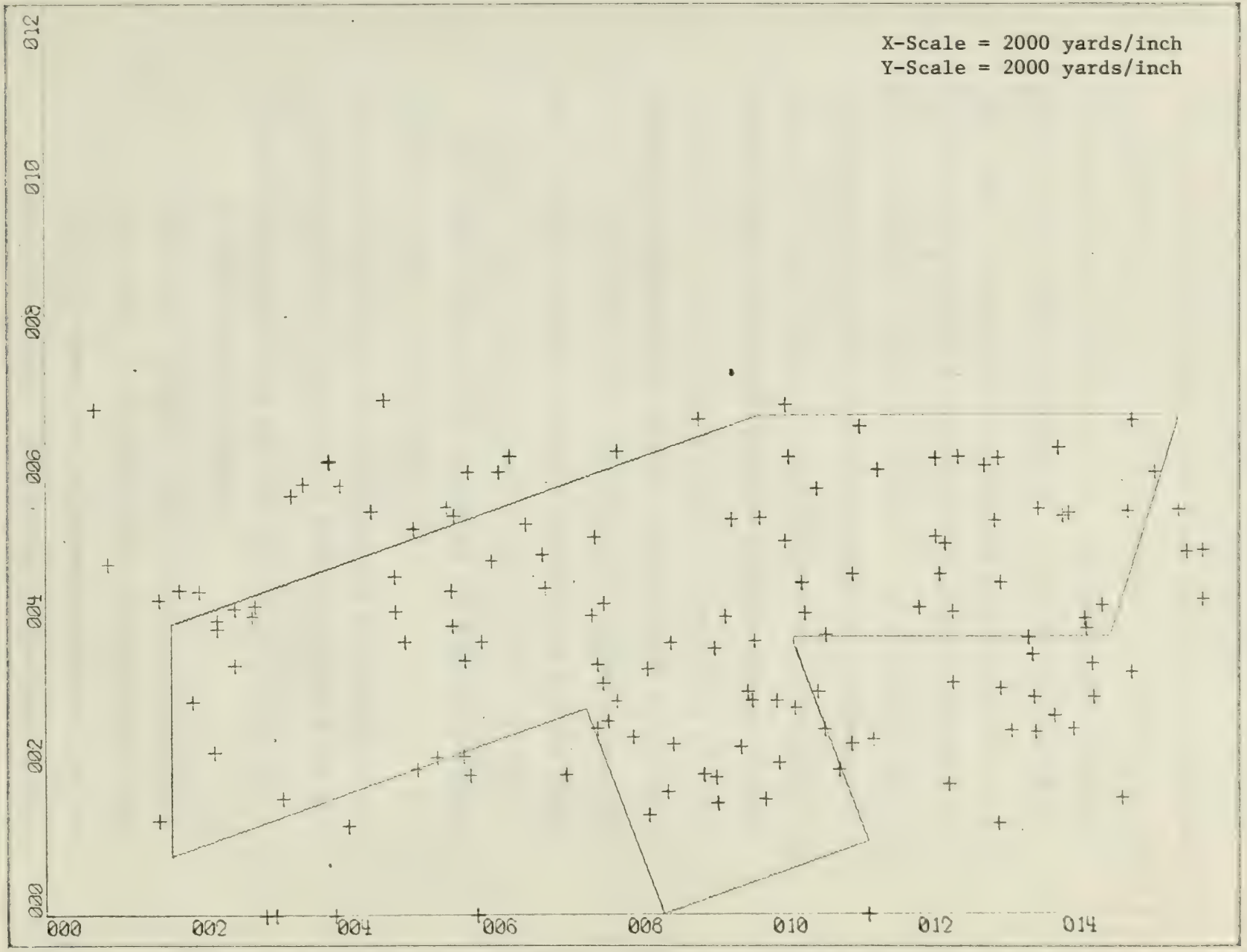



Figure 14. Randomly Generated Minefield Utilizing Distributions on the Axes.

axis would be required. It would appear that the number required is dependent on the minefield shape and the accuracy desired.

Regardless of the extent of the above approach, the basic problem of describing the "most probable" minefield remains unsolved. Each particular mine with its resultant parameters contributes in its own way to the minefield effectiveness and this has been ignored. Therefore, the mine position distribution approach is limited in application to those minefields consisting of similar type mines and where the physical environment of the mine is unimportant.

In view of the foregoing a procedure based on average minefield characteristics is recommended as an improved approximation to a "most probable" minefield. This procedure involves the selection of a sample number of runs from a large number of runs for threat assessment. While the results of no single run of program MINDEL could be considered truly representative of the "most probable" minefield, a sample consisting of those runs having characteristics approximating the expected or average characteristics of a large number of runs may be considered adequate for threat assessment purposes. For example, the Summary of Results in Appendix XIII shows for the test run of 80 iterations that the average number of mines layed was 137 of the 167 intended. Using the criterion of selecting the fields for which the number of mines layed was within plus or minus two of the average number of mines layed, runs number 11, 13, 16, 17, 34, 45, 50, 52, and 53 are selected. If the same criterion is applied

to the number of effective mines, runs number 11, 16, 34, and 45 are eliminated. Again applying this criterion to the last three statistics in turn, run number 17 becomes the primary candidate for threat assessment. To obtain a larger sample size requires only relaxing the selection criterion accordingly. Having thus determined the minefield(s) to be used for threat assessment, the user has available in the output of program MINDEL1 the mine parameters of each mine layed in the selected run(s). Without these parameters, i.e., case depth, actuation sensitivity, actuation time, and mine type, in addition to the final mine position, the vehicle, mine, and bottom characteristics are not adequately represented.

It is concluded that the average values for the statistics shown in the Summary of Results Table (MINDEL1) provide the user with a convenient guide to the selection of the particular runs for threat assessment. While this is sampling via the "back-door", it circumvents the complexity of determining a true "most probable" minefield and yet preserves the consideration of individual mines in threat assessment.

Without further investigation into the area of defining the "most probable" minefield and in lieu of a more sophisticated minefield classification technique, it is recommended that the average statistics approach be adopted; since it represents a significant improvement over existing minefield generation techniques.

## 5. Conclusions and Recommendations.

It is concluded that:

1. The Mine Delivery Model provides the U. S. Navy with the capability of simulating specific mining plans.

2. Certain features incorporated in this model, i.e., bottom contour and bottom type, provide new flexibility in minefield generation.

3. Vehicle characteristics, particularly attrition and navigational capabilities, have decided effects on the final minefield.

4. The generation of minefields for threat assessment by simple probabilities for mine positions has limited applicability.

It is recommended that:

1. This model be adopted for use by the mine warfare modeling community.

2. The limitations of the use of simple probability distributions in generating mine positions for minefield threat assessment be recognized, and that this method be used accordingly.

3. Further investigation be conducted into the feasibility of classifying the "most probable" minefield.

BIBLIOGRAPHY


1. Bowditch, N. American Practical Navigator. Hydrographic Office Publication No. 9. Washington: U. S. Government Printing Office, 1958.

2. Bureau of Naval Weapons. NAVWEPS OP 1935 (Volume 1). Mine, Underwater, Mk 27 Mods 2 & 3: Vehicle Preparation and Maintenance Instructions (U). Washington: Department of the Navy, 1964. (Confidential)

3. _____. NAVWEPS OP 2363 (Volume 1). Mine, Underwater, Mk 27 Mods 4 & 5: Vehicle Preparation and Maintenance Instructions (U). Washington: Department of the Navy, 1965.(Confidential)

4. _____. NAVWEPS OP 2637 (Volume 1). Operational Characteristics of U. S. Naval Mines: Operational Considerations in Mine Warfare (U). Washington: Department of the Navy, 1961. (Confidential)

5. _____. NAVWEPS OP 2637 (Volume 2). Operational Characteristics of U. S. Naval Mines: Aircraft Planted Mines (U). Washington: Department of the Navy, 1961. (Confidential)

6. _____. NAVWEPS OP 2637 (Volume 3). Operational Characteristics of U. S. Naval Mines: Submarine Planted Mines (U). Washington: Department of the Navy, 1961. (Confidential)

7. _____. NAVWEPS OP 2637 (Volume 4). Operational Characteristics of U. S. Naval Mines: Surface Craft Planted Mines (U). Washington: Department of the Navy, 1961. (Confidential)

8. Clark, R. E., Kubik, R. N., and Phillips, L. P. "Orthogonal Polynomial Least Squares Surface Fit. Algorith 164," Communications of the ACM, VI (April, 1963), pp. 162-163.

9. Forsythe, G. E. "Generation and Use of Orthogonal Polynomials for Data Fitting with a Digital Computer," Journal of the SIAM, V (June, 1957), pp. 74-88.

10. Hill, J. C., Utegaard, T. F., and Riorden, G. (comp.). Dutton's Navigation and Piloting. Annapolis, Maryland: U. S. Naval Institute, 1959.

11. Office of the Chief of Naval Operations. NWIP 23-2(B). Submarine Support Operations (U). Washington: Department of the Navy, 1965. (Confidential)

12. _____. NWIP 26-1. Minefield Planning (U). Washington: Department of the Navy, 1953. (Confidential)

13. _____. NWP 26(A). Mining Operations (U). Washington: Department of the Navy, 1962. (Confidential)

14. Parzen, E. Modern Probability Theory and Its Applications. New York: John Wiley and Sons, Inc., 1960.

15. Pike, M. C. and Hill, I. D. "Algorithm 266," Communications of the ACM, VIII (October, 1965), pp. 605-606.

16. Underwood, J. L. A Critical Survey of PERT/Cost, with Emphasis on the Monte Carlo Technique of Network Calculation. Monterey, California: U. S. Naval Postgraduate School, 1964.

17. U. S. Naval Mine Defense Laboratory. Report 245. The Significance of Buried Mines in Mine Warfare Systems Planning (U). Panama City, Florida: Department of the Navy, 1965.

# APPENDIX I

## PROGRAM GRID

One of the first steps in the process of generating data for the Mine Delivery Model is the planning of the physical characteristics of the minefield. This requires the determination of the geographical limits of the field, the geographical position of each mine which is to be layed in the field, and the geographical positions of reference points for the minelaying vehicles. Depending upon the size of the minefield (the number of mines to be layed in the minefield) the generation of this data, even in the form of latitude and longitude coordinates, can be a very time consuming process.

Although there is no way to avoid manual plotting of the minefield on a chart and recording the coordinates of each desired position, the user is relieved of the task of converting these coordinates to the more usable form of an X-Y grid measured in yards. This conversion is accomplished by program GRID and the converted data is stored on magnetic tape for later use.

GRID has been written as a separate main program primarily because it uses several thousand storage locations which would otherwise have to be allocated to program MINDEL. Secondly, the use of GRID as a separate program results in more efficient use of computer time when running parametric analyses with the main program (MINDEL).

Program GRID accepts input data on punched cards as latitude and longitude coordinates recorded to the nearest one-hundredth of a minute and converts this data to X-Y grid coordinates measured in yards from an inputed reference position (XREF, YREF). The converted data (all mine positions, reference positions, and field coordinates) is stored on magnetic tape and also printed out for the user as a visual check against errors in the input data.

A list of inputs for program GRID is provided in the following pages along with information pertaining to how this data is punched on the data cards. A description of how ACLDL, SHLDL, and SULDL are derived is given below.

The reference position for the conversion process is determined by enclosing the minefield and all vehicle reference points by a rectangle of whatever size is necessary. The point XREF, YREF is then the lower left hand corner of this rectangle. In other words, XREF, YREF is the origin of a cartesian coordinate system positioned so that the minefield and all reference points fall within the first quadrant.

This program will handle all geographical situations (i.e., the minefield straddles the equator, the prime meridian, or the 180 degree meridian) except the case where XREF or the longitude of a mine position falls exactly on the 180 degree meridian.

All north latitudes and east longitudes must be inputed as positive values. All south latitudes and west longitudes

must be inputed as negative values.

The use of input data recorded to the nearest one-hundredth of a minute results in an accuracy of approximately plus or minus twenty yards in any converted position. It is felt that this is an acceptable accuracy figure in view of the user's inability to obtain more accurate position information from the average nautical charts used for minefield planning purposes.

In the conversion process the inputed data is first converted to degrees and fractions of degrees and then converted to yards using the following equations[2], where each equation is divided by 0.9144 to convert from meters to yards.

Y = YLAT/57.29578

YINT = Y Latitude - YREF

XINT = X Longitude - XREF

To convert latitude:

$$Y(yds) = \frac{((111132.09-566.05*COSF(2.0*Y)+1.20*COSF(4.0*Y))*YINT}{0.9144}$$

To convert longitude:

$$X(yds) = \frac{((111415.10*COSF(Y)-94.55*COSF(3.0*Y)-.12*COSF(5.0*Y))*XINT}{0.9144}$$

If the user does not desire to utilize all three types of vehicles for laying a minefield, GRID will convert the desired

[2]N. Bowditch, American Practical Navigator (U. S. Navy Hydrographic Office Publication No. 9, 1958), p. 1187.

data if the appropriate parameters are inputed as zeroes. For example: if only aircraft and ships are to be used, NSUR, NSUC, and NMSU must be inputed as zeroes.

ACLDL, SHLDL, and SULDL are the primary inputs to program GRID, and the desired data for these matrices must be carefully recorded. Figures 15 and 16 illustrate the suggested formats for preparing this data prior to punching it on cards.

The following discussion relates to the preparation of data for ACLDL and/or SHLDL (reference figure 15).

Assume that the minefield and all mines have been plotted. Due to the large number of mines per leg shown here, it is reasonable to assume that these 271 mines are to be delivered by ships. Assume, also, that the legs have been determined, as well as the initial and terminal points for each leg (each leg must have an IP and a TP). There are 18 legs (column 1) to be transited by whatever number of ships have been assigned to the minelaying mission (one vehicle can transit more than one leg - the only limit to the number of legs per vehicle is the number of mines that the vehicle can carry). The number of mines to be layed on each leg is entered in column two. The X-coordinate of the IP for each leg is entered in column three and the Y-coordinate of each leg IP is entered in column four.

The number of the first mine to be layed on each leg is entered in column five, and this is followed by the X-coordinate and Y-coordinate of the intended position of that mine in columns six and seven. This sequence of "mine number, X-coordinate,

# HIRCRAFT OR SHIP LEG DESCRIPTION LIST

| Leg No. | No of Mines | Initial Position X Coord. | Initial Position Y Coord. | Mine No. | Intended Position X Coord. | Intended Position Y Coord. | Mine No. | Intended P X Coord. | Intended Position X Coord. | Intended Position Y Coord. | Mine No. | Intended Position X Coord. | Intended Position Y Coord. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 30 | -121.4517 | 36.4106 | 1 | -121.4889 | 36.4106 | 2 | -121.4930 | -121.5210 | 36.4106 | 30 | -121.5264 | 36.4106 |
| 2 | 31 | -121.5289 | 36.4089 | 31 | -121.5249 | 36.4089 | 32 | -121.5203 | -121.4916 | 36.4089 | | -121.4902 | 36.4089 |
| 3 | 14 | -121.4888 | 36.4070 | 60 | -121.4096 | 36.4080 | 61 | -121.4949 | | | | | |
| 4 | 21 | -121.5150 | 36.4048 | 74 | -121.5124 | 36.4048 | 75 | -121.5096 | | | | | |
| 5 | 7 | -121.4856 | 36.4029 | 95 | -121.4890 | 36.4029 | 96 | -121.4921 | | | | | |
| 6 | 17 | -121.5004 | 36.4 | | | | | | | | | | |

| | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 16 | 24 | -121.5216 | 36.3814 | 244 | -121.5194 | 36.3814 | 245 | -121.5182 | 36.3814 | 246 | -121.5167 | 36.3814 | 247 | -121.5142 | 36. |
| 17 | 3 | -121.4695 | 36.3802 | 268 | -121.4648 | 36.3802 | 269 | -121.4681 | 36.3802 | 270 | -121.4698 | 36.3802 | | -121.4712 | 36.3 |
| 18 | 1 | -121.4704 | 36.3756 | 271 | -121.4650 | 36.3756 | | -121.4642 | 36.3756 | | | | | | |
| | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | |

figure 15.

Y-coordinate," is continued across each row from column five on until all mines and their positions for each particular leg have been entered (the maximum number of mines which can be layed on a leg is 31 without a modification to the program). Following the last mine position, the X-coordinate and Y-coordinate of the leg terminal point are recorded under the next "intended position" columns with the column for the mine number left blank. This procedure is illustrated in rows 2, 17, and 18 of Figure 15.

The procedure for recording data for the SULDL matrix (reference Figure 16) is similar to that above with the exception of the firing position. For self-propelled mines the firing position coordinates and the mine intended position coordinates must be different, while they will be the same for mines not self-propelled. When recording terminal point data for SULDL, the X-coordinate and Y-coordinate of the terminal point are recorded in the firing position columns with the mine number column again left blank. This procedure is illustrated in row three of Figure 16. The maximum number of mines which can be layed on one submarine leg is 18, but this value can be increased by a program change. Note that of the mines shown in Figure 16, mines number 65, 66,and 81 are self-propelled.

Figures 15 and 16 also illustrate the numerical format for the data which is to be punched on cards. The latitude value, for example, of 36 degrees and 25.48 minutes is punched as 36.2548. The same format is used for longitude values. The negative values shown in Figures 15 and 16 indicate West longitude as explained above.

# Submarine Leg Description List

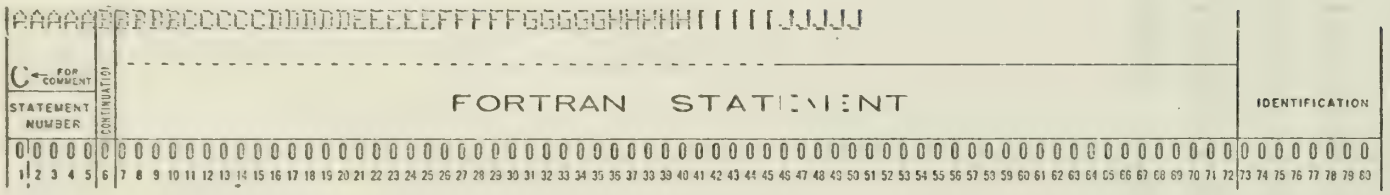
| Leg No. | No. of Mines | Initial Position | | Mine No. | Firing Position | | Intended Position | | Mine No. | Firing Position | | Intended Position | | M. N. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | X Coord. | Y Coord. | | X Coord. | Y Coord. | X Coord. | Y Coord. | | X Coord. | Y Coord. | X Coord. | Y Coord. | |
| 1 | 11 | -121.5409 | 36.4070 | 65 | -121.5361 | 36.3976 | -121.5340 | 36.3829 | 66 | -121.5342 | 36.3935 | -121.5316 | 36.3784 | 6 |
| 2 | 5 | -121.5201 | 36.3918 | 76 | -121.5226 | 36.3909 | -121.5226 | 36.3909 | 77 | -121.5252 | 36.3899 | -121.5252 | 36.3899 | 7 |
| 3 | 1 | -121.5382 | 36.4154 | 81 | -121.5362 | 36.4118 | -121.5378 | 36.3917 | | -121.5312 | 36.3992 | | | |
| 4 | 4 | -121.5292 | 36.3938 | 82 | -121.5262 | 36.3935 | -121.5262 | 36.3935 | 83 | -121.5242 | 36.3933 | -121.5242 | 36.3933 | 8 |
| 5 | 7 | -121.5211 | 36.3931 | 86 | -121.5240 | 36.3945 | -121.5240 | 36.3945 | 87 | -121.525 | | | | |

Figure 16

Immediately preceding the program listing which appears at the end of this appendix is presented a list of variables and input preparation guidelines for program GRID. The guidelines consist of a description of each data card in the order they should appear in the data deck. For each card a sample FORTRAN card strip punched with letters is shown followed by four columns; the first column is the letter key corresponding to the letters on the card strip and the second column is the corresponding FORTRAN specifications succeeded by the variable names as they appear in the program and a brief definition.

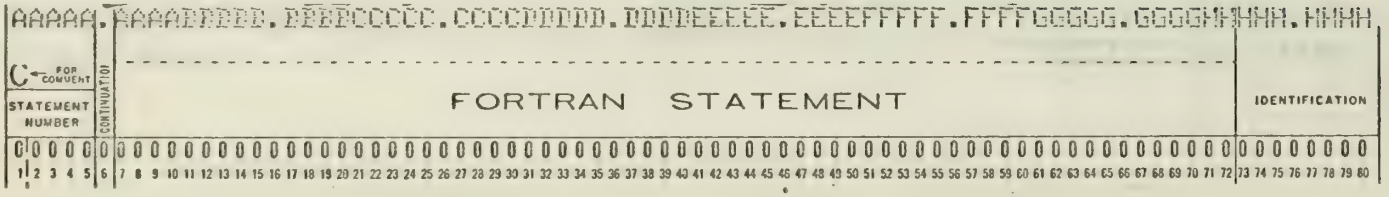
PROGRAM GRID - INPUT VARIABLES

FIRST DATA GROUP (ONE CARD)



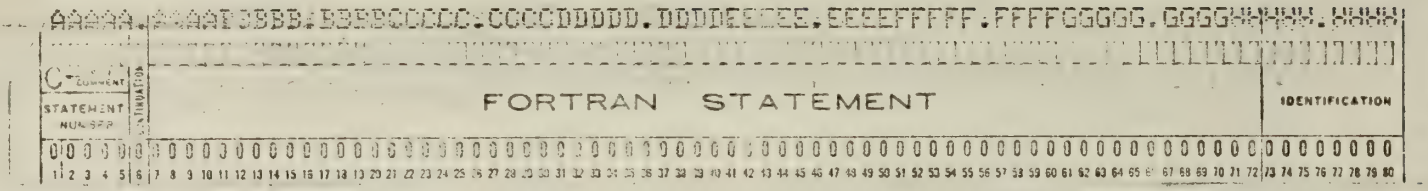

| | | | |
|---|---|---|---|
| A. | I5 | NF | THE NUMBER OF POINTS DEFINING THE MINEFIELD PERIMETER (MUST BE THE SAME AS THE NUMBER OF ENTRIES IN FXPTS AND FYPTS). |
| B. | I5 | NACR | THE NUMBER OF ROWS IN ACLDL. |
| C. | I5 | NACC | THE NUMBER OF COLUMNS IN ACLDL. |
| D. | I5 | NSHR | THE NUMBER OF ROWS IN SHLDL. |
| E. | I5 | NSHC | THE NUMBER OF COLUMNS IN SHLDL. |
| F. | I5 | NSUR | THE NUMBER OF ROWS IN SULDL. |
| G. | I5 | NSUC | THE NUMBER OF COLUMNS IN SULDL. |
| H. | I5 | NMA | THE NUMBER OF AIRCRAFT LAYED MINES. |
| I. | I5 | NMS | THE NUMBER OF SHIP LAYED MINES. |
| J. | I5 | NMSU | THE NUMBER OF SUBMARINE LAYED MINES. |

SECOND DATA GROUP (MAXIMUM OF 3 CARDS)



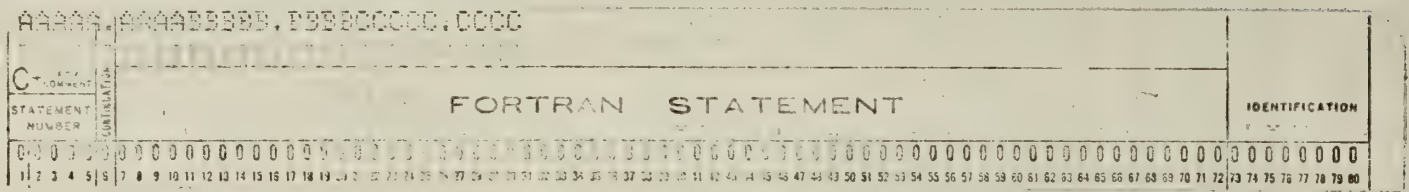

A-H. F10.4 FXPTS FIELD POINT VALUES - ELEMENTS OF FXPTS MATRIX.

THIRD DATA GROUP (MAXIMUM OF 3 CARDS)



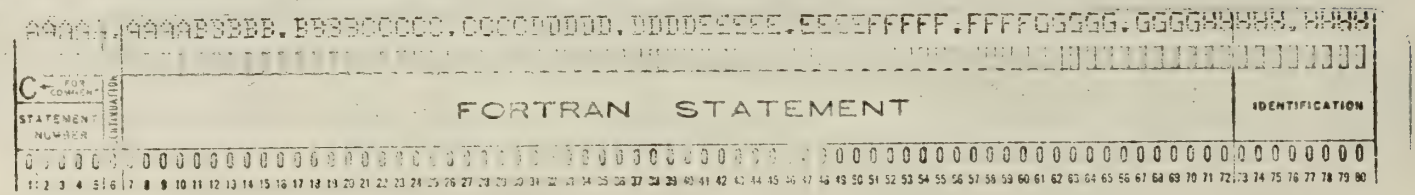

A-H. F10.4 FYPTS FIELD POINT VALUES - ELEMENTS OF FYPTS MATRIX.

FOURTH DATA GROUP (ONE CARD)



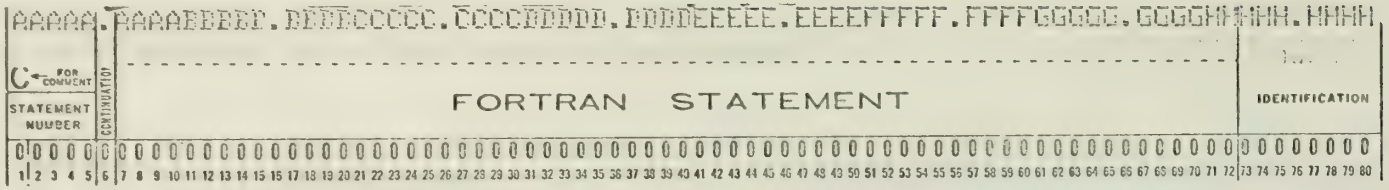

A. F10.4 XREF THE X-COORDINATE OF THE MINEFIELD REFERENCE POINT.
B. F10.4 YREF THE Y-COORDINATE OF THE MINEFIELD REFERENCE POINT.
C. F10.4 YLAT THE MEAN LATITUDE OF THE MINEFIELD.

FIFTH DATA GROUP (MAXIMUM OF 250 CARDS)



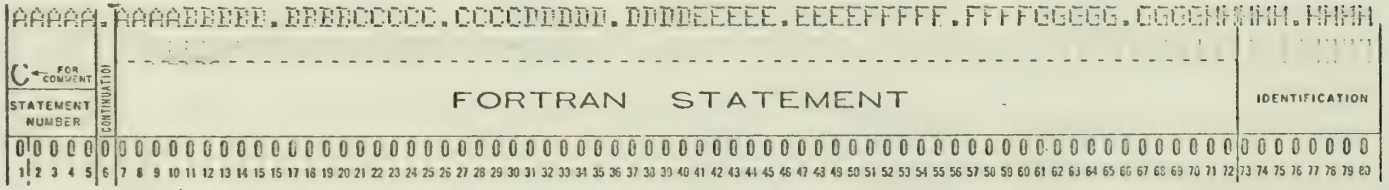

A-H. F10.4 ACLDL ELEMENTS OF ACLDL MATRIX ENTERED BY COLUMNS. I.E, THIS DATA MUST BE ENTERED AS ((ACLDL(I,J),I=1,NACR), J=1,NACC).

SIXTH DATA GROUP (MAXIMUM OF 250 CARDS)




A-H. F10.4 SHLDL ELEMENTS OF SHLDL MATRIX ENTERED BY COLUMNS. I.E, THIS DATA MUST BE ENTERED AS ((SHLDL(I,J),I=1,NSHR), J=1,NSHC).

SEVENTH DATA GROUP (MAXIMUM OF 250 CARDS)




A-H. F10.4 SULDL ELEMENTS OF SULDL MATRIX ENTERED BY COLUMNS. I.E, THIS DATA MUST BE ENTERED AS ((SULDL(I,J),I=1,NSUR), J=1,NSUC).

```
      PROGRAM GRID                                                      0001
      DIMENSION ACLDL(20,100),SHLDL(20,100),SULDL(20,100),FXPTS(20),FYPT 0002
     1S(20),XREF(1),YREF(1),YLAT(1),XIP(20),YIP(20),XTP(20),YTP(20),XPTS 0003
     2(620),YPTS(620),XFP(360),YFP(360)                                 0004
      COMMON ACLDL,SHLDL,SULDL,XPTS,YPTS,XFP,YFP                        0005
      READ 5,NF,NACR,NACC,NSHR,NSHC,NSUR,NSUC,NMA,NMS,NMSU              0006
    5 FORMAT(10I5)                                                      0007
      READ 6,(FXPTS(I),I=1,NF)                                          0008
    6 FORMAT(8F10.4)                                                    0009
      READ 7,(FYPTS(I),I=1,NF)                                          0010
    7 FORMAT(8F10.4)                                                    0011
      READ 8,XREF,YREF,YLAT                                             0012
    8 FORMAT(3F10.4)                                                    0013
      READ 9,((ACLDL(I,J),I=1,NACR),J=1,NACC)                           0014
    9 FORMAT(8F10.4)                                                    0015
      READ 10,((SHLDL(I,J),I=1,NSHR),J=1,NSHC)                          0016
   10 FORMAT(8F10.4)                                                    0017
      READ 11,((SULDL(I,J),I=1,NSUR),J=1,NSUC)                          0018
   11 FORMAT(8F10.4)                                                    0019
      MFLAG=0                                                           0020
      CALL DEGCON(FXPTS,NF)                                             0021
      CALL DEGCON(FYPTS,NF)                                             0022
      CALL DEGCON(XREF,1)                                               0023
      PRINT 26,XREF                                                     0024
   26 FORMAT(24H X REFERENCE POSITION IS,F10.3/)                        0025
      CALL DEGCON(YREF,1)                                               0026
      PRINT 27,YREF                                                     0027
   27 FORMAT(24H Y REFERENCE POSITION IS,F10.3/)                        0028
      CALL DEGCON(YLAT,1)                                               0029
      PRINT 29,YLAT                                                     0030
   29 FORMAT(17H MEAN LATITUDE IS,F10.3//)                              0031
      PRINT 40                                                          0032
   40 FORMAT(62H THE FOLLOWING COLUMN TITLES PERTAIN TO ALL DATA WHICH F 0033
     1OLLOWS//)                                                         0034
      PRINT 41                                                          0035
   41 FORMAT(13H POSITION NO.,9X,8H X COORD,7X,8H Y COORD//)            0036
```

```
      PRINT 45                                                          0037
   45 FORMAT(18H FIELD COORDINATES//)                                   0038
      CALL CONVER(FXPTS,FYPTS,XREF(1),YREF(1),NF,YLAT(1),MFLAG)         0039
      PRINT 97                                                          0040
   97 FORMAT(31H AIRCRAFT LEG INITIAL POSITIONS//)                      0041
      CALL CHANGE1(ACLDL,NACR,XREF(1),YREF(1),YLAT(1),XIP,YIP,MFLAG)    0042
      PRINT 96                                                          0043
   96 FORMAT(29H AIRCRAFT LEG TERMINAL POINTS//)                        0044
      CALL CHANGE2(ACLDL,NACC,NACR,XREF(1),YREF(1),YLAT(1),NMA,XIP,YIP,M 0045
     1FLAG)                                                             0046
      PRINT 95                                                          0047
   95 FORMAT(27H SHIP LEG INITIAL POSITIONS//)                          0048
      CALL CHANGE1(SHLDL,NSHR,XREF(1),YREF(1),YLAT(1),XIP,YIP,MFLAG)    0049
      PRINT 80                                                          0050
   80 FORMAT(25H SHIP LEG TERMINAL POINTS//)                            0051
      CALL CHANGE2(SHLDL,NSHC,NSHR,XREF(1),YREF(1),YLAT(1),NMS,XIP,YIP,M
     1FLAG)                                                             0052
      PRINT 620                                                         0053
  620 FORMAT(//32H SUBMARINE LEG INITIAL POSITIONS //)                  0054
      CALL CHANGE1(SULDL,NSUR,XREF(1),YREF(1),YLAT(1),XIP,YIP,MFLAG)    0055
      K = 1                                                             0056
      L = 1                                                             0057
      M = NSUC - 1                                                      0058
      DO 670 I = 1,NSUR                                                 0059
      DO 680 J = 6,M,5                                                  0060
      IF(SULDL(I,J+1))681,681,682                                       0061
  681 XTP(L) = SULDL(I,J)                                               0062
      L = L + 1                                                         0063
      GO TO 670                                                         0064
  682 XFP(K)=SULDL(I,J)                                                 0065
      XPTS(K) = SULDL(I,J+2)                                            0066
  680 K = K + 1                                                         0067
  670 CONTINUE                                                          0068
      K = 1                                                             0069
      L = 1                                                             0070
      DO 770 I = 1,NSUR                                                 0071
      DO 780 J = 7,NSUC,5                                               0072
```

```
      IF(SULDL(I,J-2))781,781,782                                       0073
  781 YTP(L)=SULDL(I,J)                                                 0074
      L = L + 1                                                         0075
      GO TO 770                                                         0076
  782 YFP(K)=SULDL(I,J)                                                 0077
      YPTS(K) = SULDL(I,J+2)                                            0078
  780 K = K + 1                                                         0079
  770 CONTINUE                                                          0080
      CALL DEGCON(XTP,NSUR)                                             0081
      CALL DEGCON(YTP,NSUR)                                             0082
      PRINT 785                                                         0083
  785 FORMAT(30H SUBMARINE LEG TERMINAL POINTS //)                      0084
      CALL CONVER(XTP,YTP,XREF(1),YREF(1),NSUR,YLAT(1),MFLAG)           0085
      CALL DEGCON(XPTS,NMSU)                                            0086
      CALL DEGCON(YPTS,NMSU)                                            0087
      CALL DEGCON(XFP,NMSU)                                             0088
      CALL DEGCON(YFP,NMSU)                                             0089
      MFLAG=1                                                           0090
      CALL CONVER(XPTS,YPTS,XREF(1),YREF(1),NMSU,YLAT(1),MFLAG)         0091
      CALL CONVER(XFP,YFP,XREF(1),YREF(1),NMSU,YLAT(1),MFLAG)           0092
      PRINT 790                                                         0093
  790 FORMAT(//20H SUBMARINE MINE DATA //)                              0094
      PRINT 791                                                         0095
  791 FORMAT(13H POSITION NO.,4X,15HFIRING POSITION,8X,22HMINE INTENDED 0096
     1POSITION /14X,47H XCOORD       YCOORD       XCOORD       YCOORD  0097
     2//)                                                               0098
      DO 800 K=1,NMSU                                                   0099
      PRINT 810, K,XFP(K),YFP(K),XPTS(K),YPTS(K)                        0100
  810 FORMAT(4X,I3,6X,F11.3,F11.3,5X,F11.3,F11.3)                       0101
  800 CONTINUE                                                          0102
      DO 870 I=1,NSUR                                                   0103
      SULDL(I,3) = XIP(I)                                               0104
  870 SULDL(I,4) = YIP(I)                                               0105
      L = 1                                                             0106
      K = 1                                                             0107
      M = NSUC-1                                                        0108
```

```
      DO 820 I=1,NSUR                                                    0109
      DO 830 J=6,M,5                                                     0110
      IF(SULDL(I,J-1))831,831,832                                        0111
  831 SULDL(I,J)=XTP(L)                                                  0112
      L = L + 1                                                          0113
      GO TO 820                                                          0114
  832 SULDL(I,J)=XFP(K)                                                  0115
      SULDL(I,J+2)=XPTS(K)                                               0116
      K = K + 1                                                          0117
  830 CONTINUE                                                           0118
  820 CONTINUE                                                           0119
      L = 1                                                              0120
      K = 1                                                              0121
      DO 840 I=1,NSUR                                                    0122
      DO 850 J=7,NSUC,5                                                  0123
      IF(SULDL(I,J-2))851,851,852                                        0124
  851 SULDL(I,J)=YTP(L)                                                  0125
      L = L + 1                                                          0126
      GO TO 840                                                          0127
  852 SULDL(I,J)=YFP(K)                                                  0128
      SULDL(I,J+2)=YPTS(K)                                               0129
      K = K + 1                                                          0130
  850 CONTINUE                                                           0131
  840 CONTINUE                                                           0132
      WRITE TAPE 2,NF,NACR,NACC,NSHR,NSHC,NSUR,NSUC,XREF,YREF,(FXPTS(I),  0133
     1I=1,NF),(FYPTS(I),I=1,NF),((ACLDL(I,J),I=1,NACR),J=1,NACC),((SHLDL  0134
     2(I,J),I=1,NSHR),J=1,NSHC),((SULDL(I,J),I=1,NSUR),J=1,NSUC)         0135
      END                                                                0136
```

```
      SUBROUTINE CHANGE1(A,N,P,Q,R,XIP,YIP,MM)                          0137
      DIMENSION A(20,100),XIP(20),YIP(20)                               0138
      DO 46 J=1,N                                                       0139
   46 XIP(J)=A(J,3)                                                     0140
      DO 47 K=1,N                                                       0141
   47 YIP(K)=A(K,4)                                                     0142
      CALL DEGCON(XIP,N)                                                0143
      CALL DEGCON(YIP,N)                                                0144
      CALL CONVER(XIP,YIP,P,Q,N,R,MM)                                   0145
      RETURN                                                            0146
      END                                                               0147
```

```
      SUBROUTINE CHANGE2(A,N1,N2,P,Q,R,M1,XIP,YIP,MMM)                  0148
      DIMENSION A(20,100),XTP(20),YTP(20),XPTS(620),YPTS(620),XIP(20),YI 0149
     1P(20)                                                             0150
      K=1                                                               0151
      L=1                                                               0152
      M=N1-1                                                            0153
      DO 70 I=1,N2                                                      0154
      DO 80 J=6,M,3                                                     0155
      IF(A(I,J-1))81,81,82                                              0156
   81 XTP(L)=A(I,J)                                                     0157
      L=L+1                                                             0158
      GO TO 70                                                          0159
   82 XPTS(K)=A(I,J)                                                    0160
   80 K=K+1                                                             0161
   70 CONTINUE                                                          0162
      K=1                                                               0163
      L=1                                                               0164
      DO 170 I=1,N2                                                     0165
      DO 180 J=7,N1,3                                                   0166
      IF(A(I,J-2))181,181,182                                           0167
  181 YTP(L)=A(I,J)                                                     0168
      L=L+1                                                             0169
      GO TO 170                                                         0170
  182 YPTS(K)=A(I,J)                                                    0171
  180 K=K+1                                                             0172
  170 CONTINUE                                                          0173
      CALL DEGCON(XTP,N2)                                               0174
      CALL DEGCON(YTP,N2)                                               0175
      CALL CONVER(XTP,YTP,P,Q,N2,R,MMM)                                 0176
      PRINT 500                                                         0177
  500 FORMAT(15H MINE POSITIONS//)                                      0178
      CALL DEGCON(XPTS,M1)                                              0179
      CALL DEGCON(YPTS,M1)                                              0180
      CALL CONVER(XPTS,YPTS,P,Q,M1,R,MMM)                               0181
      DO 400 I=1,N2                                                     0182
      A(I,3)=XIP(I)                                                     0183
```

```
400 A(I,4)=YIP(I)                                                    0184
    L=1                                                              0185
    K=1                                                              0186
    M=N1-1                                                           0187
    DO 410 I=1,N2                                                    0188
    DO 420 J=6,M,3                                                   0189
    IF(A(I,J-1))421,421,422                                          0190
421 A(I,J)=XTP(L)                                                    0191
    L=L+1                                                            0192
    GO TO 410                                                        0193
422 A(I,J)=XPTS(K)                                                   0194
    K=K+1                                                            0195
420 CONTINUE                                                         0196
410 CONTINUE                                                         0197
    L=1                                                              0198
    K=1                                                              0199
    DO 510 I=1,N2                                                    0200
    DO 520 J=7,N1,3                                                  0201
    IF(A(I,J-2))521,521,522                                          0202
521 A(I,J)=YTP(L)                                                    0203
    L=L+1                                                            0204
    GO TO 510                                                        0205
522 A(I,J)=YPTS(K)                                                   0206
    K=K+1                                                            0207
520 CONTINUE                                                         0208
510 CONTINUE                                                         0209
    RETURN                                                           0210
    END                                                              0211
```

```
      SUBROUTINE DEGCON(A,N)                                          0212
      DIMENSION A(620)                                                0213
      DO 100 L=1,N                                                    0214
      IF(A(L))101,100,102                                             0215
  101 LFLAG=0                                                         0216
      GO TO 110                                                       0217
  102 LFLAG=1                                                         0218
  110 CALL FRACT(A,L,LFLAG)                                           0219
  100 CONTINUE                                                        0220
      RETURN                                                          0221
      END                                                             0222
```

```
      SUBROUTINE FRACT(A,M,LF)                                          0223
      DIMENSION A(620)                                                  0224
      A(M)=ABSF(A(M))                                                   0225
      LA=A(M)                                                           0226
      FLPT=LA                                                           0227
      REM=DIMF(A(M),FLPT)                                               0228
      ADD=REM/0.6                                                       0229
      A(M)=FLPT+ADD                                                     0230
      IF(LF)115,115,116                                                 0231
  115 A(M)=-A(M)                                                        0232
  116 RETURN                                                            0233
      END                                                               0234
```

```
      SUBROUTINE CONVER(A,B,C,D,N,YDEG,M)                                0235
      DIMENSION A(620),B(620)                                            0236
      Y=YDEG/57.29578                                                    0237
      DO 5 I=1,N                                                         0238
      YINT=B(I)-D                                                        0239
      B(I)=((111132.09-566.05*COSF(2.0*Y)+1.20*COSF(4.0*Y))*YINT)/.9144  0240
    5 CONTINUE                                                           0241
    6 DO 7 J=1,N                                                         0242
      IF(C) 10,11,11                                                     0243
   10 XINT=A(J)-C                                                        0244
      GO TO 20                                                           0245
   11 IF(A(J))10,13,13                                                   0246
   13 DXDT=180.000+A(J)                                                  0247
      X=180.000-C                                                        0248
      XINT=DXDT+X                                                        0249
   20 A(J)=((111415.10*COSF(Y)-94.55*COSF(3.0*Y)-.12*COSF(5.0*Y))*XINT   0250
     1)/.9144                                                            0251
    7 CONTINUE                                                           0252
      IF(M)50,50,60                                                      0253
   50 DO 30 K=1,N                                                        0254
      PRINT 21,K,A(K),B(K)                                               0255
   21 FORMAT(4X,I3,12X,F11.3,6X,F11.3)                                   0256
   30 CONTINUE                                                           0257
   60 RETURN                                                             0258
      END                                                                0259
```

## APPENDIX II

## PROGRAM SURFIT

The final position of a mine is of great importance to this model not only because the determination of this position is one of the primary objectives of the model, but because it plays a vital role in determining other characteristics of the mine.

When any mine is layed its characteristics are dependent upon (1) whether it is operable or not, and (2), if it is operable, its physical environment. The question of mine operability is, among other things, decided by whether or not the mine is damaged on bottom impact. This in turn is a function of the type of bottom the mine impacts upon. The question of the physical environment of the mine is decided by whether or not the mine buries and the final water depth of the mine case.

In order to provide a ready reference of data for answering these questions over the continuum of possible final mine positions in the minefield area program SURFIT is used. This program fits, in the least squares sense, a polynomial function, Z, of two independent variables, X and Y, to values of the dependent variable specified at points on a rectangular grid in the plane of the independent variables. That is, given a set of values of Z at points of the X, Y grid, the program finds the polynomial

$$Z = \sum_{n=0}^{N} \sum_{m=0}^{M} A_{n,m} X^n Y^m$$ which best fits the data.

The optimum values of N and M as well as the coefficients, $A_{n,m}$, are determined.

The same rectangle as was used to define the minefield area in program GRID is recommended for use here. This rectangle is then divided into a grid of 45 or less segments on each axis. The fineness of this grid should be determined by the user as necessary to adequately describe both the water depth variations and the bottom type variations (as explained in Appendix III).

When the water depth and bottom type data are prepared as described in the "Input Data" section, program SURFIT generates the coefficients of the depth contour and the bottom type contour polynomials. This information, along with the degrees of the independent variables in each polynomial, is written on magnetic tape for use in program MINDEL. A complete printout of all input data, the generated coefficients, and a table of differences is provided. The table of differences provides the user with the necessary information to determine the accuracy of the fit in each case. It should be noted that in the case of the depth contour any land masses within the rectangle must be given a negative depth gradient if an accurate fit is to be obtained.

Program SURFIT[3] was copied from the U. S. Naval Postgraduate School Computer Facility Library and modified slightly for use in this model. The program listing which appears at the end of this appendix is basically the original listing, although not all aspects of it are used. Specifically, that part of the program

[3]R. E. Clark, R. N. Kubik, and L. P. Phillips, "Orthogonal Polynomial Least Squares Surface Fit, Algorithm 164." Communications of the ACM, VI (April, 1963), pp. 162-163.

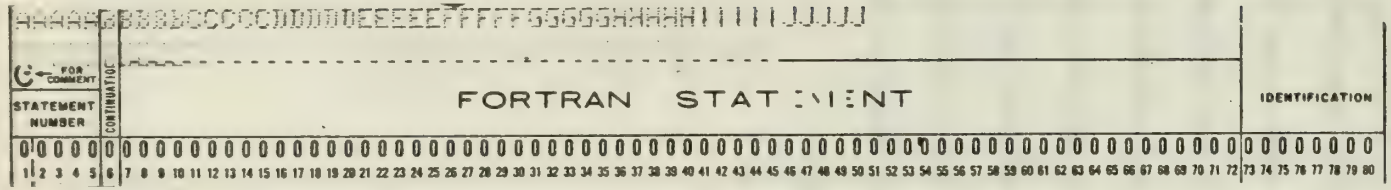
which pertains to new X and Y values that can be read in for evaluation after the polynomial has been calculated is not used in this program. The basic computational technique is used, however, in program MINDEL where it is included as Subroutine Depth.

The "Input Data" section which immediately follows describes the complete range of input data for program SURFIT as a convenience to the user. The description lists the punched cards in the order they are to appear in the data deck. For each card there is a sample FORTRAN card strip punched in the correct format with letters. In the four columns below each sample card strip may be found the letter keys corresponding to the letters on the card strip, the corresponding FORTRAN specifications, the variable names as they appear in the program, and a brief definition. The input variables which have been specifically used in this adaptation of program SURFIT are denoted by asterisks.

PROGRAM SURFIT

INPUT DATA

1. FIRST DATA CARD(*)




| | | | |
|---|---|---|---|
| A. | I5 | NMAX | THE MAXIMUM DEGREE OF X TO BE CONSIDERED(LESS THAN OR EQUAL TO IMAX). |
| B. | I5 | MMAX | THE MAXIMUM DEGREE OF Y TO BE CONSIDERED(LESS THAN OR EQUAL TO JMAX). |
| C. | I5 | IMAX | THE NUMBER OF X VALUES(LESS THAN OR EQUAL TO 45). |
| D. | I5 | JMAX | THE NUMBER OF Y VALUES(LESS THAN OR EQUAL TO 45). |
| E. | I5 | IEQX | A CONTROL PARAMETER. IF IEQX=0, ALL X VALUES WILL BE READ IN (SEE(2) SECOND DATA GROUP FOR FORMAT) IF IEQX=1,ONLY THE FIRST AND LAST VALUE OF X WILL BE READ IN AND THE PROGRAM GENERATES THE REMAINING EQUALLY SPACED X VALUES (SEE (3) SECOND DATA GROUP FOR FORMAT). |
| F. | I5 | IEQY | A CONTROL PARAMETER WITH THE SAME FUNCTION AS IEQX FOR Y VARIABLES(SEE (4) AND (5) THIRD DATA CARD/GROUP FOR FORMAT). |
| G. | I5 | KWT | A CONTROL PARAMETER. IF KWT=0, ALL WEIGHTS CORRESPONDING TO X AND Y VALUES ARE UNITY. IF KWT=1, THE U(I),WEIGHTS EACH CORRESPONDING TO A X(I), WILL BE READ IN, AND ALL WEIGHTS CORRESPONDING TO Y VALUES ARE CONSIDERED UNITY. IF KWT=2,THE W(J),WEIGHTS EACH CORRESPONDING TO A Y(J) WILL BE READ IN AND ALL WEIGHTS CORRESPONDING TO X(J) ARE CONSIDERED UNITY. IF KWT=3,THE U(I) AND W(J), REPRESENTING THE WEIGHTS CORRESPONDING TO X(I) AND Y(J) WILL BE READ IN(SEE FOURTH |

DATA GROUP FOR FORMAT).

H. I5 IFMT A CONTROL PARAMETER. IF IFMT=0, THEN THE FORMAT USED TO READ Z VALUES IS ASSUMED TO BE(4F20.0). IF IFMT=1, VARIABLE FORMAT CARD WILL BE SUPPLIED BY THE USER(SEE(10) FIFTH DATA CARD).

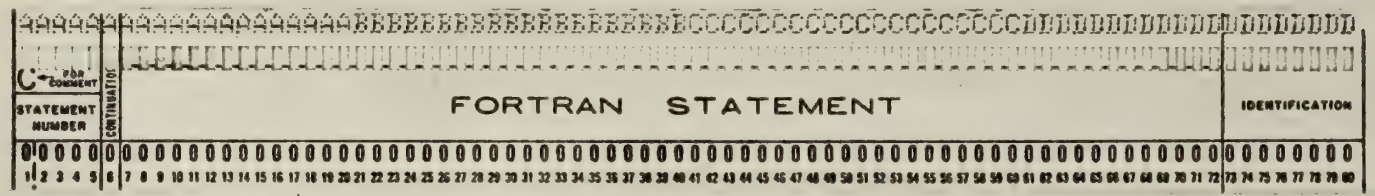
I. I5 NEWCTL A CONTROL PARAMETER. IF NEWCTL=0, THEERE ARE NO NEW X AND Y VALUES TO BE READ IN FOR EVALUATION AFTER THE POLYNOMIAL HAS BEEN FOUND. IF NEWCTL=1, NEW X AND Y VALUES WILL BE READ IN FOR EVALUATION(SEE (12),(13) SEVENTH,EIGHTH DATA GROUP FOR FORMAT).

J. I5 KODE A CONTROL PARAMETER. IF KODE=1, AN ADDITIONAL COEFFICIENT TABLE OF ORTHOGONAL POLYNOMIALS WILL BE PRINTED. FOR REGULAR OUTPUT KODE=0.

NOTE-ALL FIXED POINT INTEGERS MUST BE RIGHT JUSTIFIED IN THEIR RESPECTIVE FIELDS

IF IEQX=0, THE FOLLOWING CARD(S) WILL BE READ.

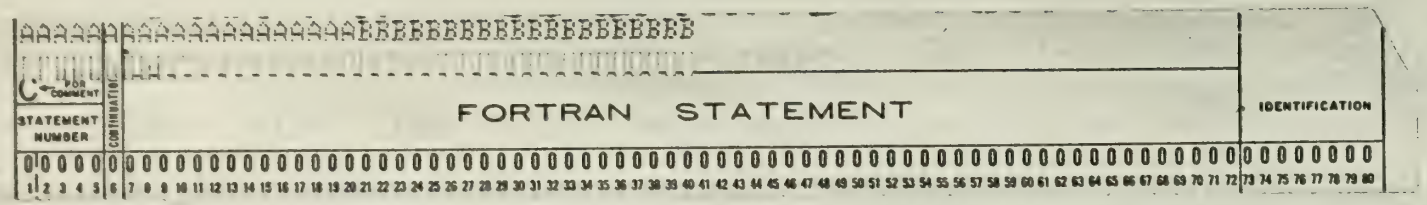
2. SECOND DATA GROUP (MAXIMUM OF 12 CARDS)




A-D. F20.0 X(I) IMAX ORIGINAL X VALUES.

IF IEQX=1, THE FOLLOWING CARD WILL BE READ.

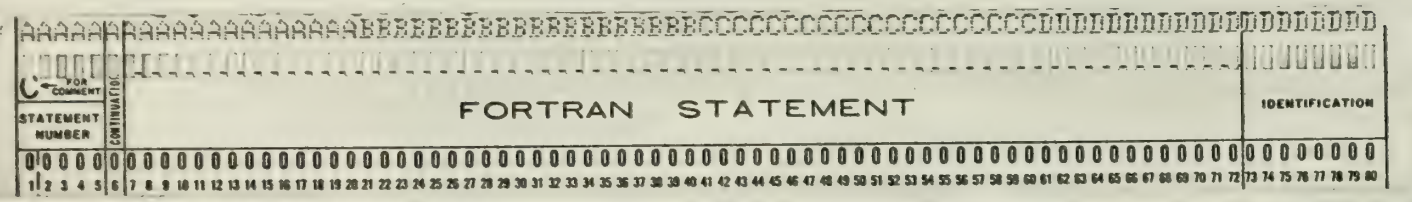
3. SECOND DATA CARD (*)




A. F20.0 XFIRST THE SMALLEST VALUE OF X.
B. F20.0 XLAST THE LARGEST VALUE OF X.

IF IEQY=0, THE FOLLOWING CARD(S) WILL BE READ.

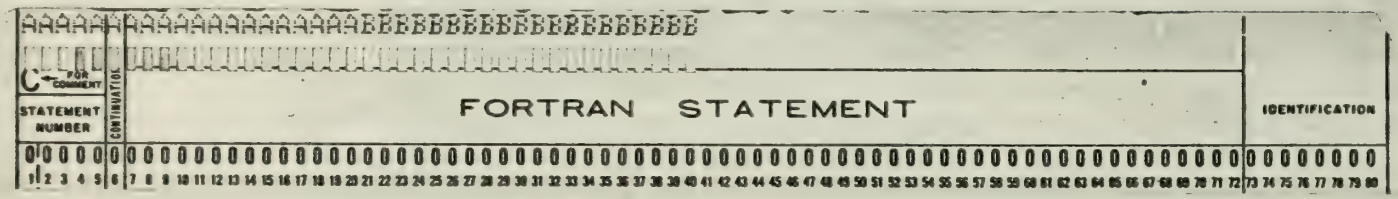
4. THIRD DATA GROUP (MAXIMUM OF 12 CARDS)




A-D. F20.0 Y(J) JMAX ORIGINAL Y VALUES.

IF IEQY=1, THE FOLLOWING CARD WILL BE READ.

5. THIRD DATA CARD (*)



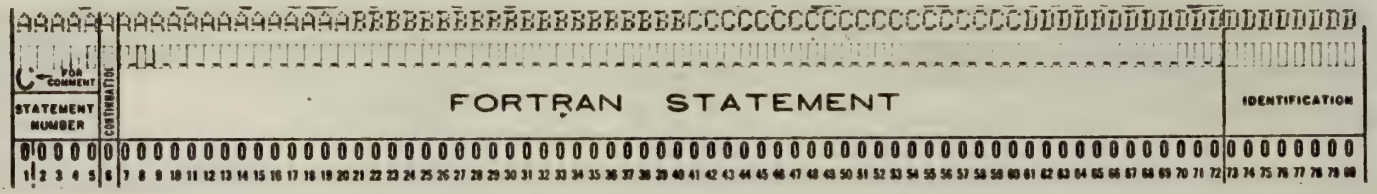

A. F20.0 YFIRST THE SMALLEST VALUE OF Y.

B. F20.0 YLAST THE LARGEST VALUE OF Y.

IF KWT=0, NO CARD WILL BE READ FOR THIS GROUP (*)

IF KWT=1, THE FOLLOWING CARD(S) WILL BE READ.

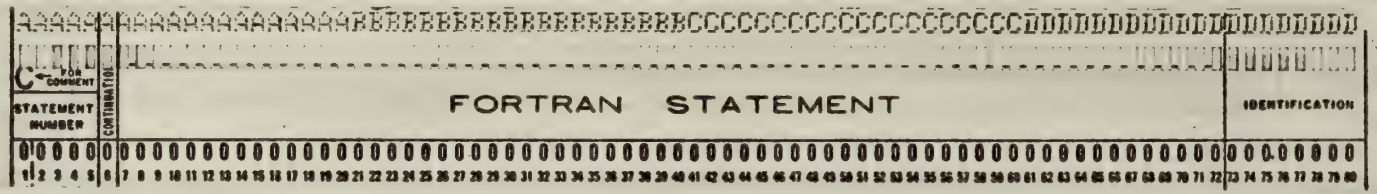
6. FOURTH DATA GROUP (MAXIMUM OF 12 CARDS)




A-D. F20.0 U(I) IMAX NUMBERS OF WEIGHTS CORRESPONDING TO THE X(I).

IF KWT=2, THE FOLLOWING CARD(S) WILL BE READ.

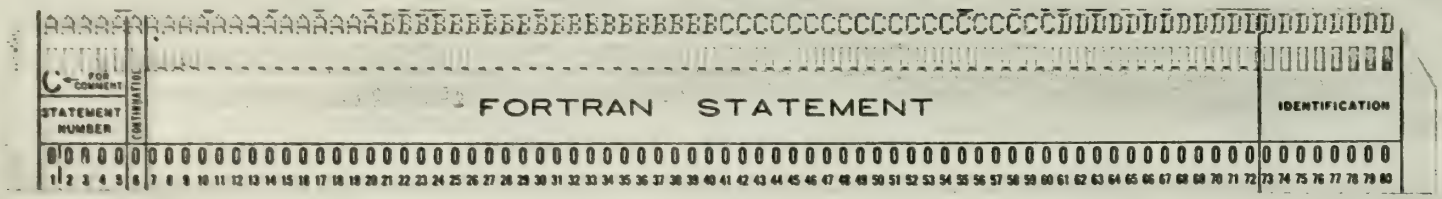
7. FOURTH DATA GROUP (MAXIMUM OF 12 CARDS)




A-D. F20.0 W(J) JMAX NUMBERS OF WEIGHTS CORRESPONDING TO THE Y(J).

IF KWT=3, THE FOLLOWING CARD(S) WILL BE READ.

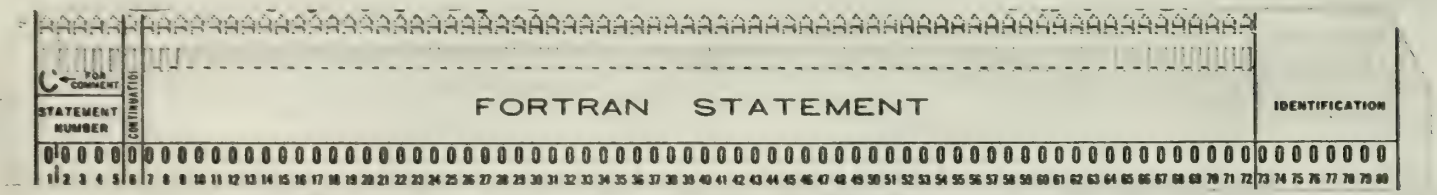
8. FOURTH DATA GROUP (MAXIMUM OF 24 CARDS)




A-D. F20.0 U(I), W(J) IMAX PLUS JMAX NUMBERS OF WEIGHTS CORRESPONDING TO THE X(I) AND Y(J) VALUES.

IF IFMT=1, THE FOLLOWING CARD WILL BE READ.

9. FIFTH DATA CARD.



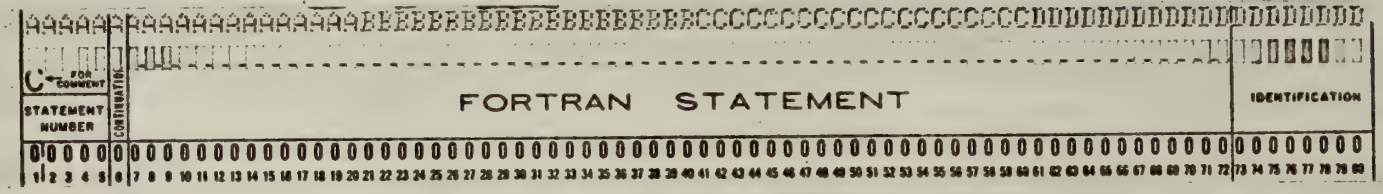

A. 10A8 IFMTZ A VARIABLE FORMAT CARD USED TO READ IN Z VALUES(SEE (10) SIXTH DATA GROUP).

10. SIXTH DATA GROUP

Z(I,J) IMAX TIMES JMAX ORIGINAL Z VALUES MUST APPEAR ON THE NEXT CARDS UNTIL (IMAX TIMES JMAX) OF THEM HAVE BEEN READ USING THE VARIABLE FORMAT SUPPLIED BY THE USER. THE Z MATRIX SHOULD BE PUNCHED ROW-WISE.

IF IFMT=0, THE FOLLOWING CARDS WILL BE READ.

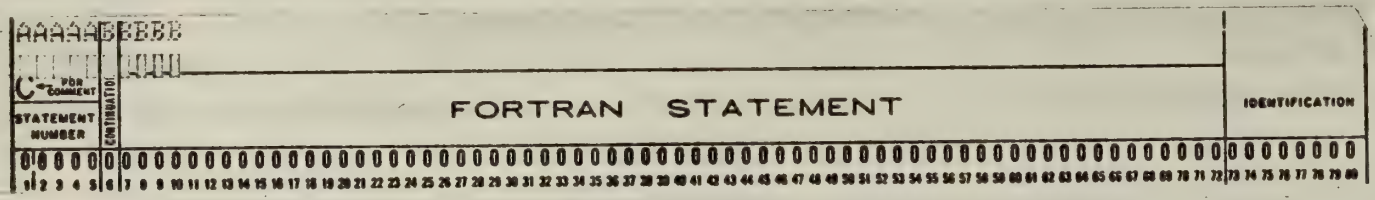
11. SIXTH DATA GROUP (MAXIMUM OF 507 CARDS) (*)




A-D. F 2.0 Z(I,J) (IMAX TIMES JMAX) ORIGINAL Z VALUES MUST APPEAR ON THE NEXT CARDS UNTIL (IMAX TIMES JMAX) OF THEM HAVE BEEN READ. THE Z MATRIX SHOULD BE PUNCHED ROW-WISE, USING AS MANY CARDS AS NECESSARY, WITHOUT INTERMEDIATE BLANK FIELDS. Z(1,1),Z(1,2), ...,Z(1,JMAX),Z(2,1),Z(2,2),...,Z(2,JMAX),...,Z(IMAX,1), Z(IMAX,2),...,Z(IMAX,JMAX).

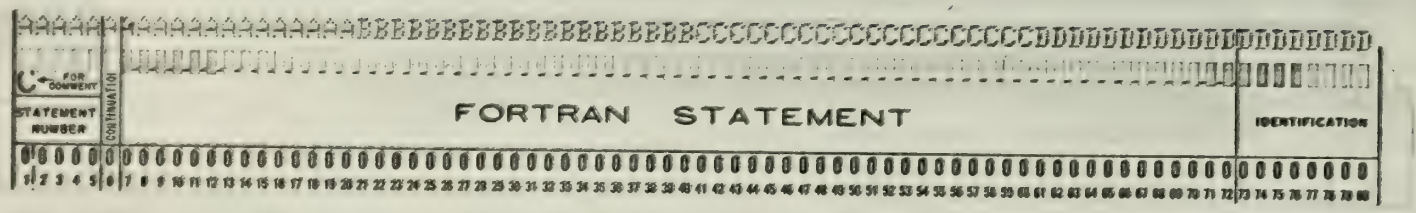
THE FOLLOWING CARDS ARE READ ONLY IF NEWCTL=1.

12. SEVENTH DATA CARD




| | | | |
|---|---|---|---|
| A. | I5 | NEWX | THE NUMBER OF NEW X VALUES(LESS THAN OR EQUAL TO 45). |
| B. | I5 | NEWY | THE NUMBER OF NEW Y VALUES(LESS THAN OR EQUAL TO 45). |

13. EIGHTH DATA GROUP




A-D. F20.0 X(I), Y(J) NEWX PLUS NEWY NUMBERS OF NEW X VALUES AND NEW Y VALUES MUST APPEAR ON THE NEXT CARDS.

```
      PROGRAM SURFIT                                                    0000   0
      DIMENSION X(45),U(45),Y(45),W(45),Z(45,45),IFMTZ(10)              00000010
      DIMENSION FXPTS(10),FYPTS(10),ACLDL(8,31),SHLDL(7,70),SULDL(4,37) 00000015
C   NMAX=(DEGREE OF X) +1,MMAX=(DGR OF Y)+1, IMAX= NO.OF X ,JMAX= NO.OF Y00000020
C   KODE=0 MINIMUM PRINTED OUTPUT , KODE=1 MAX PRINTED OUTPUT , KWT =0  00000030
C   ALL WEIGHTS =1 , KWT =1 READ IMAX NUMBERS OF U(I) WEIGHTS FOR X ,THEN00000040
C   READ JMAX NUMBERS OF W(J) WEIGHTS FOR Y.                            00000050
C   NEWCTL =1  READ NEXT CONTROL CARD FOR NEW X,Y TO BE READ IN FOR     00000060
C   EVALUATION AFTER THE SURFACE HAS BEEN FOUND. NEWCTL=0 NO NEW XANDY  00000070
      REWIND 2                                                          000   71
      READ TAPE 2, NF,NACR,NACC,NSHR,NSHC,NSUR,NSUC,XREF,YREF,(FXPTS(I),00000072
     1I=1,NF),(FYPTS(I),I=1,NF),((ACLDL(I,J),I=1,NACR),J=1,NACC),((SHLDL00000073
     2(I,J),I=1,NSHR),J=1,NSHC),((SULDL(I,J),I=1,NSUR),J=1,NSUC)        00000074
      REWIND 2                                                          000   75
      WRITE TAPE 2,NF,NACR,NACC,NSHR,NSHC,NSUR,NSUC,XREF,YREF,(FXPTS(I),00000076
     1I=1,NF),(FYPTS(I),I=1,NF),((ACLDL(I,J),I=1,NACR),J=1,NACC),((SHLDL00000077
     2(I,J),I=1,NSHR),J=1,NSHC),((SULDL(I,J),I=1,NSUR),J=1,NSUC)        00000078
      DO9999 INUM=1,2                                                   00000079
      READ 50,NMAX,MMAX,IMAX,JMAX,IEQX,IEQY,KWT,IFMT ,NEWCTL,KODE       00000080
   50 FORMAT(10I5)                                                      00000 90
      MMAX=MMAX+1                                                       00000100
      NMAX=NMAX+1                                                       00000110
      IF(IEQX)  61,63,61                                                00000120
C   IEQX=0,ALL XS VALUE TO BE READIN ,IEQX=1, ONLY FIRST AND LAST VALUES 00000130
C   OF XS READ IN                                                       00000140
   61 READ 51, XFIRST, XLAST                                            00000150
   51 FORMAT (4F20.0)                                                   00000160
      XNUM = IMAX-1                                                     00000170
      DELTAX = (XLAST-XFIRST) /XNUM                                     00000180
      II=IMAX-1                                                         00000190
      X(1)= XFIRST                                                      00000200
      DO 62 I=2,II                                                      00000210
   62 X(I)= X(I-1)+DELTAX                                               00000220
      X(IMAX) = XLAST                                                   00000230
      GO TO 64                                                          00000240
   63 READ 51,(X(I),I=1,IMAX)                                           00000250
```

```
   64 IF (IEQY) 71,73,71                                                    00000260
   71 READ 51,YFIRST,YLAST                                                  00000270
      YNUM = JMAX-1                                                         00000280
      DELTAY = (YLAST -YFIRST)/ YNUM                                        00000290
      JJ= JMAX-1                                                            00000300
      Y(1) =YFIRST                                                          00000310
      DO 72  J=2,JJ                                                         00000320
   72 Y(J)= Y(J-1)+DELTAY                                                   00000330
      Y(JMAX)= YLAST                                                        00000340
      GO TO 74                                                              00000350
   73 READ 51,(Y(J),J=1,JMAX)                                               00000360
   74 IF (KWT)  75,79,75                                                    00000370
   75 GO TO (76,77,78),KWT                                                  00000380
C  KWT=0,NO WGT,=1 RDX WGT , =2, Y WGT , =3  X AND Y WGT                    00000390
   76 READ 51,(U(I),I=1,IMAX)                                               00000400
      DO 176 J=1,JMAX                                                       00000410
  176 W(J)=1.                                                               0000 420
      GO TO 80                                                              00000430
   77 READ 51,(W(J),J=1,JMAX)                                               00000440
      DO 177 I=1,IMAX                                                       00000450
  177 U(I)=1.                                                               00000460
      GO TO 80                                                              00000470
   78 READ 51,(U(I),I=1,IMAX)                                               00000480
      READ 51,(W(J),J=1,JMAX)                                               00000490
      GO TO 80                                                              0000 500
   79 DO 179 I=1,IMAX                                                       00000510
  179 U(I)=1.                                                               00000520
      DO 279 J=1,JMAX                                                       00000530
  279 W(J)=1.                                                               00000540
C  IF IFMT =0,USE REGULAR FORMAT,IF IFMTZ=1,READ1 VARIABLE FORMAT CARD      00000550
   80 IF (IFMT )  81,82,81                                                  00000560
   82 READ 51, ((Z(I,J),J=1,JMAX),I=1,IMAX)                                 00000570
      GO TO 90                                                              00000580
C  Z ALWAYS READ IN ROW WISE                                                00000590
   81 READ 181,IFMTZ                                                        00000600
  181 FORMAT (10A8)                                                         00000610
```

```
      READ IFMTZ, ((Z(I,J),J=1,JMAX),I=1,IMAX)                          00000620
   90 CALL SURFACE (X,U,Y,W,Z,NMAX,MMAX,IMAX,JMAX,          CMINSQD,     00000630
     1CSUMDIF,CMAXDIF       ,NEWCTL,KODE)                              00000640
                                                                       00000645
 9999 CONTINUE                                                         000  650
      END
```

```
      SUBROUTINE SURFACE(X,U,Y,W,Z,NMAX,MMAX,IMAX,JMAX,          CMINSQD,00000660
     1CSUMDIF,CMAXDIF      ,NEWCTL,KODE)                                 00000670
      DIMENSION DENPA(45),DENQA(45),ALPHA(45,45),P(45,45),Q(45,45),      00000680
     1PC(45,45),QC(45,45),A(45),B(45),C(45),D(45),X(45),U(45),Y(45),     00000690
     2W(45),Z(45,45),BETA(45,45),PHI(45,45),ZCOMP(45,45)                 00000700
     3,IFMTZ(10)                                                         00000710
     4,ZDIFF(45,45)                                                      00000720
      COMMON P,Q                                                         00000730
      EQUIVALENCE(P,PC),(Q,QC) ,(ALPHA,ZCOMP)                            00000740
      IF (IMAX-8)    3001,3001,3002                                      00000750
 3001 IF (JMAX-16)  3003,3003,3004                                       00000760
 3003 ICON= 1H0                                                          00000770
      GO TO 3010                                                         00000780
 3004 ICON =1H1                                                          00000790
      GO TO 3010                                                         00000800
 3002 IF (IMAX-16)  3005,3005,3004                                       00000810
 3005 IF (JMAX-10)  3003,3003,3004                                       00000820
C     PRINT INPUT DATA X,Y,AND Z                                         00000830
 3010 PRINT 2100                                                         0000 840.
 2100 FORMAT(1H120X48HINPUT DATA INDEPENDENT  VARIABLES  X (I)           00000850
     1   ///)                                                           0000 860
      PRINT2101,( X(I), I=1,IMAX )                                       00000870
 2101 FORMAT        (8E14.7)                                            00000880
      PRINT 2102                                                         00000890
 2102 FORMAT( /// 20X 45H INPUT DATA INDEPENDENT  VARIABLES  Y (J)       00000900
     1   ///)                                                           0000 910
      PRINT2101,(Y(J),J=1,JMAX)                                          00000920
      PRINT 3000                                                         0000 930
 3000 FORMAT (///)                                                       00000940
      PRINT 2103,ICON                                                    00000950
 2103 FORMAT(A1,20X         40HINPUT DATA  DEPENDENT  VARIABLES Z(I,J)   00000960
     1   /// )                                                          0000 970
      PRINT2105,(J, J=1,JMAX)                                            00000980
 2105 FORMAT(4H I/J I13,7I14 / (4X,I13,7I14))                            00000990
      DO2106 I=1,IMAX                                                    00001000
 2106 PRINT2107,I,(Z(I,J),J=1,JMAX)                                      00001010
```

```
 2107 FORMAT(// I3,8E14.7 /(3X8E14.7))                                  00001020
C NORMALIZATION OF VARIABLES                                            00001030
      SUMX=0.                                                           00001040
      SUMY=0.                                                           00001050
      SUMZ=0.                                                           00001060
      DO 10 I=1,IMAX                                                    00001070
   10 SUMX=SUMX+X(I)                                                    00001080
      XIMAX=IMAX                                                        00001090
      XMEANX= SUMX/XIMAX                                                00001100
      PRINT 1001,XMEANX                                                 00001110
 1001 FORMAT(//10H XMEANX =       E20.10)                               00001120
      DO 20 I=1,IMAX                                                    00001130
   20 X(I)=X(I)-XMEANX                                                  00001140
C     PRINT 1002, (X(I),I=1,IMAX)                                       00001150
C1002 FORMAT (5E19.10)                                                  00001160
      DO 30 J=1,JMAX                                                    00001170
   30 SUMY=SUMY+Y(J)                                                    00001180
      XJMAX=JMAX                                                        00001190
      YMEANY=SUMY/XJMAX                                                 00001200
      PRINT 1003, YMEANY                                                00001210
 1003 FORMAT(//10H YMEANY =           E20.10)                           00001220
      DO 40 J=1,JMAX                                                    00001230
   40 Y(J)=Y(J)-YMEANY                                                  00001240
C     PRINT 1004, (Y(J),J=1,JMAX)                                       00001250
C1004 FORMAT(5E19.10)                                                   00001260
      DO 50 I=1,IMAX                                                    00001270
      DO 50 J=1,JMAX                                                    00001280
   50 SUMZ=SUMZ+Z(I,J)                                                  00001290
      XIJMAX=IMAX*JMAX                                                  00001300
      ZMEANZ=SUMZ/XIJMAX                                                00001310
      PRINT 1005,ZMEANZ                                                 00001320
 1005 FORMAT(//10H ZMEANZ =   E20.10)                                   00001330
      DO 60 I=1,IMAX                                                    00001340
      DO 60 J=1,JMAX                                                    00001350
   60 Z(I,J)=Z(I,J)-ZMEANZ                                              00001360
C EVALUATE ORTHOGONAL POLYNOMIALS                                       00001370
```

```
      XNUMA=0.                                                          00001380
      DENA=0.                                                           00001390
      DO 70 I=1,IMAX                                                    00001400
      P(1,I)=1.                                                         00001410
      XNUMA=XNUMA+U(I)*X(I)                                             00001420
   70 DENA=DENA+U(I)                                                    00001430
C     PRINT 1007, DENA,XNUMA                                            00001440
C1007 FORMAT ( 7H DENA= 10X,F10.4, 8H XNUMA= 10X,F10.4)                 00001450
      A(2)=XNUMA/DENA                                                   00001460
      DO 80  I=1,IMAX                                                   00001470
   80 P(2,I)=X(I)-A(2)                                                  00001480
C     PRINT 1008,(P(2,I),I=1,IMAX)                                      00001490
 1008 FORMAT (9H P(2,I)= 10X,3F10.4)                                    00001500
      DO 90 N=3,NMAX                                                    00001510
      XNUMA=0.                                                          00001520
      DENA=0.                                                           00001530
      DENB=0.                                                           00001540
      DO 85 I=1,IMAX                                                    00001550
      XNUMA=XNUMA+U(I)*X(I)*P(N-1,I)**2                                 00001560
      DENA=DENA+U(I)*P(N-1,I)**2                                        00001570
   85 DENB=DENB+U(I)*P(N-2,I)**2                                        00001580
C     PRINT 1085, XNUMA,DENA,DENB                                       00001590
C1085 FORMAT ( 8H XNUMA= F10.4, 7H DENA= F10.4,7H DENB= F10.4)          00001600
      A(N)=XNUMA/DENA                                                   00001610
      B(N)=DENA/DENB                                                    00001620
C     PRINT 1086, A(N),B(N)                                             00001630
C1086 FORMAT (   7H A(N)= F10.4, 7H B(N)= F10.4)                        00001640
      DO 90 I=1,IMAX                                                    00001650
   90 P(N,I)=(X(I)-A(N))*P(N-1,I)-B(N)*P(N-2,I)                         00001660
C     PRINT 1090, (P(NMAX,I),I=1,IMAX)                                  00001670
C1090 FORMAT (12H P(NMAX,I)= 3F10.4)                                    00001680
      XNUMC=0.                                                          00001690
      DENC=0.                                                           00001700
      DO 100 J=1,JMAX                                                   00001710
      Q(1,J)=1.                                                         00001720
      XNUMC=XNUMC+W(J)*Y(J)                                             00001730
```

```
  100 DENC=DENC+W(J)                                                    00001740
      C(2)=XNUMC/DENC                                                   00001750
C     PRINT 1100, (Q(1,J),J=1,JMAX)                                     00001760
C1100 FORMAT (   9H Q(1,J)= 3F10.4)                                     00001770
C     PRINT 1101, XNUMC,DENC,C(2)                                       00001780
C1101 FORMAT (  8H XNUMC= F10.4,7H DENC= F10.4,7H C(2)= F10.4)          00001790
      DO 110 J=1,JMAX                                                   00001800
  110 Q(2,J)=Y(J)-C(2)                                                  00001810
C     PRINT 1102, (Q(2,J),J=1,JMAX)                                     00001820
C1102 FORMAT (   9H Q(2,J)= 3F10.4)                                     00001830
      DO 120 M=3,MMAX                                                   00001840
      XNUMC=0.                                                          00001850
      DENC=0.                                                           00001860
      DEND=0.                                                           00001870
      DO 115 J=1,JMAX                                                   00001880
      XNUMC=XNUMC+W(J)*Y(J)*Q(M-1,J)**2                                 00001890
      DENC=DENC+W(J)*Q(M-1,J)**2                                        00001900
  115 DEND=DEND+W(J)*Q(M-2,J)**2                                        00001910
      C(M)=XNUMC/DENC                                                   00001920
      D(M)=DENC/DEND                                                    00001930
C     PRINT 1103, XNUMC,DENC,DEND,C(M),D(M)                             00001940
C1103 FORMAT (8H XNUMC= F10.4,7H DENC= F10.4,7H DEND= F10.4, 7H C(M)=    00001950
C    1F10.4, 7H D(M)= F10.4)                                            00001960
      DO 120 J=1,JMAX                                                   00001970
  120 Q(M,J)=(Y(J)-C(M))*Q(M-1,J)-D(M)*Q(M-2,J)                         00001980
C     PRINT 1104,(Q(MMAX,J),J=1,JMAX)                                   00001990
C1104 FORMAT (12H Q(MMAX,J)= 3F10.4)                                    00002000
C EVALUATE CONTRIBUTION OF EACH ORTHOGONAL POLYNOMIAL TO THE MINIMIZA-  00002010
C TION OF THE RESIDUALS                                                 00002020
      DO 130 N=1,NMAX                                                   00002030
      DENPA(N)=0.                                                       00002040
      DO 130 I=1,IMAX                                                   00002050
  130 DENPA(N)=DENPA(N)+U(I)*P(N,I)**2                                  00002060
C     PRINT 1130,(DENPA(N),N=1,NMAX)                                    00002070
C1130 FORMAT  (11H DENPA(N)= 3F10.4)                                    00002080
      DO 140 M=1,MMAX                                                   00002090
```

```
      DENQA(M)=0.                                                       00002100
      DO 140 J=1,JMAX                                                   00002110
  140 DENQA(M)=DENQA(M)+W(J)*Q(M,J)**2                                  00002120
C     PRINT 1140, (DENQA(M),M=1,MMAX)                                   00002130
C1140 FORMAT (11H DENQA(N)=        3F10.4)                              00002140
      DO 160 N=1,NMAX                                                   00002150
      DO 160 M=1,MMAX                                                   00002160
      ALPH=0.                                                           00002170
      DO 150 I=1,IMAX                                                   00002180
      DO 150 J=1,JMAX                                                   00002190
      ALPH=ALPH+U(I)*W(J)*Z(I,J)*P(N,I)*Q(M,J)                          00002200
C     PRINT 1150,N,M,I,J,ALPH                                           00002210
  150 CONTINUE                                                          00002220
C1150 FORMAT (3H N= I3,3H M= I3,3H I=I3,3H J=I3,6H ALPH=F10.4)          00002230
      ALPHA(N,M)=ALPH/(DENPA(N)*DENQA(M))                               00002240
  160 BETA(N,M)=ALPHA(N,M)*ALPH                                         00002250
      DO 1160 N=1,3                                                     00002260
C     PRINT 1161,(ALPHA(N,M),BETA(N,M),M=1,3)                           00002270
C1161 FORMAT (22H ALPHA,BETA, ALTERNATE 6F10.4)                         00002280
 1160 CONTINUE                                                          00002290
C APPLICATION OF GAUSS CRITERION TO DETERMINE THE DEGREE POLY WHICH     00002300
C  YIELDS THE CLOSEST FIT TO THE GIVEN DATA                             00002310
      SUMZSQ=0.                                                         00002320
      DO 170 I=1,IMAX                                                   00002330
      DO 170 J=1,JMAX                                                   00002340
  170 SUMZSQ=SUMZSQ+U(I)*W(J)*Z(I,J)**2                                 00002350
C     PRINT 2170,SUMZSQ                                                 00002360
C2170 FORMAT (10H  SUMZSQ = E20.10)                                     00002370
      IT=1                                                              00002390
      IS=1                                                              00002380
      DO 190 N=1,NMAX                                                   00002400
      BETASUM=0.                                                        00002410
      DO 190 M=1,MMAX                                                   00002420
      XX=IMAX*JMAX-N*M                                                  00002430
      DO 175 L=1,N                                                      00002440
  175 BETASUM=BETASUM+BETA(L,M)                                         00002450
```

```
C      PRINT 1170,BETASUM                                              00002460
C1170 FORMAT ( 10H BETASUM= F10.4          )                           00002470
       IF( BETASUM-SUMZSQ)   172,172,171                               00002480
  171 TRIGAUS=0.                                                       00002490
       GO TO 174                                                       00002500
  172 IF (IMAX*JMAX-N*M) 2172,171,2172                                 00002510
 2172 TRIGAUS=(SUMZSQ-BETASUM)/XX                                      00002520
C 173 PRINT 1173,TRIGAUS                                              00002530
C1173 FORMAT ( 9HTRIGAUS= F10.4)                                      00002540
  174 IF(N-1) 176,375,176                                             00002550
  375 IF(M-1) 176,177,176                                             00002560
  177 GAUSCRT=TRIGAUS                                                 00002570
C      PRINT 1177, GAUSCRT                                            00002580
C1177 FORMAT (  32H GAUSSCRIT EQUAL TRIALGAUSSCRIT F10.4)             00002590
       GO TO 190                                                      00002600
  176 IF(GAUSCRT - TRIGAUS) 180,178,180                               00002610
  178 IF(N*M-IS*IT) 179,180,180                                       00002620
  179 IS=N                                                            00002630
       IT=M                                                           00002640
  180 CONTINUE                                                        00002650
C      PRINT 1180, GAUSCRT,TRIGAUS,N,M,IS,IT                          00002660
 1180 FORMAT (  10H GAUSCRT= F10.4,10H TRIGAUS= F10.4,4I10)           00002670
       IF( GAUSCRT - TRIGAUS) 190,190,181                             00002680
  181 GAUSCRT=TRIGAUS                                                 00002690
       IS=N                                                           00002700
       IT=M                                                           00002710
  190 CONTINUE                                                        00002720
       PRINT 1180,GAUSCRT,TRIGAUS, N,M,IS,IT                          00002730
       NMAX=IS                                                        00002740
       MMAX=IT                                                        00002750
       TEMP=(IMAX*JMAX-NMAX*MMAX)/(IMAX*JMAX)                         00002760
       XMINSQD=SQRTF( GAUSCRT*TEMP)                                   00002770
C EVALUATION OF ORTHO POLY COEFFICIENTS                               00002780
       DO 200 N=1,NMAX                                                00002790
       PC(N,N)=1.                                                     00002800
       NM1=N-1                                                        00002810
```

```
      DO 200 IS=1,NM1                                                   00002820
      PC(N,IS) = -A(N)* PC(NM1,IS)                                      00002830
      IF(IS-1) 197,198,197                                              00002840
  197 PC(N,IS)= PC(N,IS) +PC(N-1,IS-1)                                  00002850
  198 IF(IS-NM1) 199,200,199                                            00002860
  199 PC(N,IS)=PC(N,IS) - B(N)* PC(N-2,IS)                              00002870
C 200 PRINT 1106,N,IS,PC(N,IS)                                          00002880
  200 CONTINUE                                                          00002890
C1106 FORMAT (4H N= I3,5H IS, I3, 11H PC(N,IS)= E19.10)                 00002900
      DO 210 M=1,MMAX                                                   00002910
      QC(M,M)=1.                                                        00002920
      MM1=M-1                                                           00002930
      DO 210 IT=1,MM1                                                   00002940
      QC(M,IT)= -C(M)* QC(MM1,IT)                                       00002950
      IF(IT-1)207,208,207                                               00002960
  207 QC(M,IT)= QC(M,IT)+ QC(MM1,IT-1)                                  00002970
  208 IF(IT-MM1) 209,210,209                                            00002980
  209 QC(M,IT)=QC(M,IT)-D(M)*QC(M-2,IT)                                 00002990
C 210 PRINT 1107,M,IT,Q(M,IT)                                           00003000
  210 CONTINUE                                                          00003010
C1107 FORMAT (4H M=I3,5H IT= I3, 11H QC(M,IT), E19.10)                  00003020
C EVALUATION OF APPROXIMATING ORTHO POLY COEFF                          00003030
      DO 230 IS=1,NMAX                                                  00003040
      DO 220 IT=1,MMAX                                                  00003050
      PHI(IS,IT)=0.                                                     00003060
      DO 220 N=IS,NMAX                                                  00003070
      DO 220 M=IT,MMAX                                                  00003080
  220 PHI(IS,IT)=PHI(IS,IT)+ALPHA(N,M)*PC(N,IS)*QC(M,IT)                00003090
C     PRINT 1230,(PHI(IS,IT),IT=1,MMAX)                                 00003100
C1230 FORMAT (13H PHI(IS,IT)= 3F10.4)                                   00003110
  230 CONTINUE                                                          00003120
C EVALUATE OF DEPENDENT VARIABLES USING THE APPROXIMATING ORTHO POLY    00003130
      CMINSQD=0.                                                        00003140
      CSUMDIF=0.                                                        00003150
      CMAXDIF=0.                                                        00003160
      DO 250 I=1,IMAX                                                   00003170
```

```
      DO 250 J=1,JMAX                                                    00003180
      ZCOMP(I,J)=0.                                                      00003190
  240 IS=NMAX                                                            00003200
  248 POLY=PHI(IS,MMAX)                                                  00003210
  241 IT=MMAX-1                                                          00003220
  249 POLY=POLY*Y(J)+PHI(IS,IT)                                          00003230
      IF(IT-1) 1999,247,242                                              00003240
  242 IT=IT-1                                                            00003250
      GO TO 249                                                          00003260
  247     ZCOMP(I,J)=ZCOMP(I,J)*X(I)+POLY                                00003270
C     PRINT 1108,POLY,I,J,  ZCOMP(I,J)                                   00003280
C1108 FORMAT (    7H POLY= F10.4, 4H I= I3, 4H J= I3,10HZCOMP(I,J) F10.400003290
  243 IF(IS-1) 2000,245,244                                              00003300
  244 IS=IS-1                                                            00003310
      GO TO 248                                                          00003320
  245 RESCOMP=Z(I,J)-ZCOMP(I,J)                                          00003330
      ZCOMP(I,J)=ZCOMP(I,J)+ZMEANZ                                       00003340
      CMINSQD=CMINSQD+U(I)*W(J)*RESCOMP**2                               00003350
      ABRESCP=ABSF(RESCOMP)                                              00003360
      IF(ABRESCP-CMAXDIF) 250,250,246                                    00003370
  246 CMAXDIF=ABRESCP                                                    00003380
  250 CONTINUE                                                           00003390
      PRINT 1251                                                         00003400
 1251 FORMAT(     80H1  COMPUTED Z(I,J) AS EVALUATED BY APPROXIMATING OR00003410
     1THOGONAL POLYNOMIAL                          ///)                 00003420
      PRINT 2105,(J,J=1,JMAX)                                            00003430
      DO 260 I=1,IMAX                                                    00003440
  260 PRINT 2107, I, (ZCOMP(I,J),J=1,JMAX)                               00003450
      CMINSQD=SQRTF(CMINSQD/XIJMAX   )                                   00003460
      CSUMDIF=CSUMDIF/XIJMAX                                             00003470
      IF (KODE) 261,262,261                                              00003480
  261 PRINT 1271,ICON                                                    00003490
 1271 FORMAT(A1,///55HCOEFFICIENTS OF APPROXIMATING ORTHOGONAL POLYNOMIA00003500
     1L                 //)                                             00003510
      PRINT 1272                                                         00003520
 1272 FORMAT(     7H DEGREE 30X  14H DEGREE  OF  Y//      7H OF  X    )  00003530
```

```
      NMAXM1=NMAX-1                                                    00003540
      PRINT 1273,(J,J=0,NMAXM1      )                                  00003550
      PRINT 1273,(J,J=0,NMAXM1      )                                  00003550
 1273 FORMAT( 10X,5I20)                                                00003560
      DO 300 I=1,MMAX                                                  00003570
      II=I-1                                                           00003580
      PRINT 270,II ,(PHI(I,J),J=1,NMAX)                                00003590
  270 FORMAT(//I5,5X,5E20.10/(10X,5E20.10))                            00003600
  300 CONTINUE                                                         00003610
  262 PHI(1,1)=PHI(1,1)+ZMEANZ                                         00003620
      WRITE TAPE 2,XMEANX,YMEANY,NMAX,MMAX,PHI                         00003625
      DO 4001 I=1,IMAX                                                 00003630
      DO 4001 J=1,JMAX                                                 00003640
 4001 ZDIFF(I,J)=Z(I,J)-ZCOMP(I,J)+ZMEANZ                              00003650
      PRINT 4002                                                       00003660
 4002 FORMAT( 1H1 20X, 42H TABLE OF DIFFERENCES Z(I,J)- ZCOMP(I,J)  ///)00003670
      PRINT 2105,(J,J=1,JMAX)                                          00003680
      DO 4003 I=1,IMAX                                                 00003690
 4003 PRINT 2107,I,(ZDIFF(I,J),J=1,JMAX)                               00003700
C  TO EVALUATE  NEW  DEPENDENT VARIABLES USING THE APPROX. POLY        00003710
      IF(NEWCTL)  500,501,500                                          00003720
  500 READ 502,NEWX,NEWY                                               00003730
  502 FORMAT(2I5)                                                      00003740
      READ 503,(X(I),I=1,NEWX)                                         00003750
      READ 503,(Y(J),J=1,NEWY)                                         00003760
  503 FORMAT (4F20.0)                                                  00003770
      DO 550 I=1,NEWX                                                  00003780
      DO 550 J=1,NEWY                                                  00003790
      ZCOMP(I,J)=0.                                                    00003800
      IS=NMAX                                                          00003810
  548 POLY=PHI(IS,MMAX)                                                00003820
      IT=MMAX-1                                                        00003830
  549 POLY=POLY*(Y(J)-YMEANY)+PHI(IS,IT)                               00003840
      IF(IT-1) 2000,547,542                                            00003850
  542 IT=IT-1                                                          00003860
      GO TO 549                                                        00003870
```

```
  547 ZCOMP(I,J)=ZCOMP(I,J)*(X(I)-XMEANX)+POLY                          00003880
  543 IF(IS-1)2000,550,544                                              00003890
  544 IS=IS-1                                                           00003900
      GO TO 548                                                         00003910
  550 CONTINUE                                                          00003920
C  PRINT NEW DEPENDENT VARIABLES                                        00003930
      IF(NEWX-8)  3101, 3101,3102                                       00003940
 3101 IF(NEWY-16) 3103, 3103, 3104                                      00003950
 3103 JCON =1H0                                                         00003960
      GO TO 3110                                                        00003970
 3104 JCON =1H1                                                         00003980
      GO TO 3110                                                        00003990
 3102 IF(NEWX-16) 3105,3105,3104                                        00004000
 3105 IF(NEWY-10) 3103,3103,3104                                        00004010
 3110 PRINT 551  ,JCON                                                  00004020
  551 FORMAT(A1,///50H COMPUTED Z(I,J) EVALUATED FROM NEW X AND NEW Y   00004030
     1            ///)                                                 00004040
      PRINT 2105,(J,J=1,NEWY)                                           00004050
      DO 560  I=1,NEWX                                                  00004060
  560 PRINT 2107,I,(ZCOMP(I,J),J=1,NEWY)                                00004070
  561 PHI(1,1)=PHI(1,1)-ZMEANZ                                          00004080
      CALL POLYXY(PHI,NMAX,MMAX,-XMEANX,-YMEANY)                        00004090
      PHI(1,1)=PHI(1,1)+ZMEANZ                                          00004100
C     PRINT  SCALED COEFFICIENTS ARRAY                                  00004110
      PRINT 2108                                                        00004120
 2108 FORMAT(    49H1  COEFFICIENTS OF APPROXIMATING POLYNOMIAL    /// )00004130
      PRINT 1272                                                        00004140
      PRINT 1273,(N,N=0,NMAXM1)                                         00004150
      DO 301 I=1,MMAX                                                   00004160
      II=I-1                                                            00004170
      PRINT 270,II ,(PHI(I,J),J=1,NMAX)                                 00004180
  301 CONTINUE                                                          00004190
      PRINT 1190                                                        00004200
 1190 FORMAT (////91H  MEAN SQUARED        COMPUTED MEAN SQUARED        00004210
     1MAXIMUM DIFFERENCES                                    ///)       00004220
      PRINT 1270,XMINSQD,CMINSQD,CMAXDIF                                00004230
```

```
1270 FORMAT (E15.10,5X,E20.10, 7X,E20.10///)                    00004240
     RETURN                                                     00004250
2000 PRINT 2001                                                 00004260
2001 FORMAT (  6H ERROR)                                        00004270
     RETURN                                                     00004280
1999 PRINT 1990                                                 00004290
1990 FORMAT(15H IT ERROR STOP      )                            00004300
     STOP                                                       00004310
     END                                                        00004320
```

```
      SUBROUTINE POLYXY(PHI,NMAX,MMAX,XMEANX,YMEANY)                    00004330
      DIMENSION R(45),PHI(45,45),C(45)                                  00004340
      NM1=NMAX-1                                                        00004350
      MM1=MMAX-1                                                        00004360
      DO 40 I=1,NM1                                                     00004370
   40 R(I)=-XMEANX                                                      00004380
C     FIND COEFF ARRAY C OF POLY=(X-XMEANX)**I-1                        00004390
      DO 10 I=2,NMAX                                                    00004400
      CALL COEFFN(C,R,I-1)                                              00004410
C     PRINT 200,(C(M),M=1,I )                                           00004420
  200 FORMAT(  7H COEFF   5E20.10)                                      00004430
C NEED IMPROVMENT HERE AFTER PROGRAM CHECK OUT                          00004440
      DO 10 J=1,MMAX                                                    00004450
      K=I-1                                                             00004460
   11 PHI(K,J)=PHI(K,J)+C(K)*PHI(I,J)                                   00004470
C     PRINT 201,I,J,K,PHI(K,J)                                          00004480
  201 FORMAT (3H I=I2,3H J=I2,3H K=I2,   E20.10)                        00004490
      K=K-1                                                             00004500
      IF(K)11,10,11                                                     00004510
   10 CONTINUE                                                          00004520
C     PRINT 20,((PHI(I,J),J=1,3),I=1,3)                                 00004530
   20 FORMAT(3F10.5)                                                    00004540
      DO 140 I=1,MM1                                                    00004550
  140 R(I)=-YMEANY                                                      00004560
C     FIND COEFF ARRAY C FOR Y POLY=(Y-YMEANY)**I-1                     00004570
      DO 110 J=2,MMAX                                                   00004580
      CALL COEFFN(C,R,J-1)                                              00004590
C     PRINT 200,(C(M),M=1,J )                                           00004600
      DO 110 I=1,NMAX                                                   00004610
      K=J-1                                                             00004620
  111 PHI(I,K)=PHI(I,K)+C(K)*PHI(I,J)                                   00004630
C     PRINT 201,I,J,K,PHI(K,J)                                          00004640
      K=K-1                                                             00004650
      IF(K)111,110,111                                                  00004660
  110 CONTINUE                                                          00004670
      RETURN                                                            00004680
      END                                                               00004690
```

```
      SUBROUTINE COEFFN(A,R,N)                                          00004700
      DIMENSION CT(45),S(45),A(45),R(45),CL(45)                         00004710
C      A(1)=COEFF OF CONSTANT,  A(2)=COEFF OF X,  ....A(N+1)=COEFF OF X*00004720
      CT(1)=0.                                                          00004730
      CL(1)=0.                                                          00004740
      DO 10  I=1,N                                                      00004750
      CT(I+1)=R(I)                                                      00004760
      CL(I+1)=CT(I+1)                                                   00004770
      S(I) = 0.                                                         00004780
   10 S(1) = S(1)+R(I)                                                  00004790
      A(1) = S(1)                                                       00004800
      DO 20  I=2,N                                                      00004810
   20 A(I) = A(I-1)-R(I-1)                                              00004820
      DO 40  J=2,N                                                      00004830
      DO 35 I=J,N                                                       00004840
      IM=I-1                                                            00004850
      SUM=0.                                                            00004860
      DO 31 K=1,IM                                                      00004870
   31 SUM=SUM+CL(K)                                                     00004880
   35 CT(I)=R(I-1)*SUM                                                  00004890
      DO 30  I=J,N                                                      00004900
   30 S(J)=S(J)+CL(I)*A(I)                                              00004910
      S(J)=S(J)*(-1.)**J                                                00004920
      DO 40 I=1,N                                                       00004930
   40 CL(I)=CT(I)                                                       00004940
      S(1)=(-1.)*S(1)                                                   00004950
      DO 41 I=1,N                                                       00004960
   41 A(I)=S(N+1-I)                                                     00004970
      A(N+1)=1.                                                         00004980
      RETURN                                                            00004990
      END                                                               00005000
```

## APPENDIX III

## BOTTOM TYPE CODING AND ITS APPLICATION

In examining the aspects of minelaying it becomes obvious that the events of importance should include mine burying and damage to the mine on bottom impact (the latter becomes of particular importance in aerial mining). It is also clear that the outcomes of these events are dependent on the type of bottom material at the mine's final position as well as water depth. It is assumed that for water depths greater than 20 feet the mine's sinking rate has greatly slowed and is a function of the mine's weight and dynamic drag only; hence in the aircraft and ship sequences of events the probability of damage on bottom impact is zero for water depths greater than 20 feet. In the submarine sequence, since there always exists some possibility of the mine retaining some additional velocity from tube ejection or self-propulsion, the probability of impact damage is always considered, but it is small and independent of water depth.

It is assumed that the type of bottom material can be represented by a numeric code for each of the bottom sediments in the order of their increasing consistency on a scale of zero to five as in Figure 17.

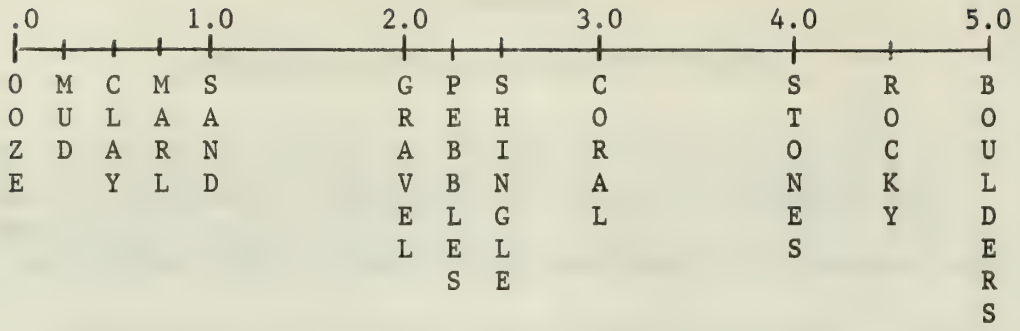



Figure 17. Scale for Bottom Material[4,5]

Further, an examination of any nautical chart demonstrates the variability of bottom material over the area of a typical minefield. To simulate the variability in bottom material a grid, preferably the same grid as used to code water depth, is superimposed over the area under consideration and each grid point is assigned a value according to the scale in Figure 17. A surface polynomial is then fitted to the bottom type values as described in Appendix II.

It is also assumed that the probability of burying and the probability of damage on bottom impact are linear functions of bottom type, i.e., the value assigned to the material at the mine's final position. The probability of burying has a maximum of the probability of burying in the softest bottom (PBSB) provided by the user for each mine type for ooze and a minimum of zero for boulders, and has the following equation:

[4]U.S. Navy Hydrographic Office, Nautical Chart Symbols and Abbreviations, Chart No. One, January, 1957, p.14.

[5]N. Bowditch, American Practical Navigator, U.S. Navy Hydrographic Office Publication No. Nine, 1958, p.108.

PB = PBSB - ( PBSB / 5.0 )( BTMTYP )

where PB is the probability of burying and BTMTYP is the value for the bottom material. The probability of bottom impact damage has a maximum of the probability of bottom impact damage on the hardest bottom (PDHB) provided by the user for each mine type for boulders and a minimum of zero for ooze, and has the following equation:

PD = ( PDHB / 5.0 )( BTMTYP )

where PD is the probability of damage on bottom impact. For example, let BTMTYP have the value, two, i.e., gravel. Assume the user has provided a value of .95 for PBSB and a value of .75 for PDHB, then

PB = .95 - ( .95 / 5.0 )( 2.0 ) = .57

PD = ( .75 / 5.0 )( 2.0 ) = .34 .

A uniform random number is then compared to these probabilities to determine whether the mine buries or not and whether the mine is damaged or not.

# APPENDIX IV

## PROGRAM LISTING, INPUT FORMAT, AND VARIABLE NAMES FOR PROGRAM MINDEL

On the succeeding pages of this appendix appear the program listing, input format, and variable names for Program MINDEL. The input format immediately follows the program listing and describes the punched card input data deck by card type. The card types are shown in the order of their appearance in the data deck along with an indication of the number of cards of each type required and their order. Further, for each card type, there is a sample FORTRAN card strip on which letters are punched according to the format required by the program. Below each card strip in four columns appear the letter keys corresponding to the letters on the card strip, the FORTRAN specifications, the variable names as they appear in the program, and brief definitions. Included in each definition and by notes where appropriate are guidelines to variable values and to the program changes required when certain values are exceeded. Following the input format section an alphabetical list of all variable names and their definitions which appear in Program MINDEL and except those defined in the input format is presented.

```
      PROGRAM MINDEL                                                    00010
      DIMENSION VAL(50,11),ACCL(10,6),ACLDL(8,31),ACMCL(24,14),XLDL(277)00020
     1,TD(135),BR(7,6),PHI(45,45),MDL(167,8),OUTPUT(30),XLCOD(6),SHCL(1 0003
     20,9),SHLDL(7,70),SHMCL(10,8),BOT(45,45),SULDL(4,37),SUCL(10,17),SU00040
     3MCL(8,13),IRMAT(100),BRTEMP(7,6),PROB(14),SMROB(20),FXPTS(10),FYPT00050
     4S(10)                                                             00055
      COMMON PHI,VAL,ACLDL,SHLDL,SULDL,MDL,BOT,ACMCL,XLDL,SUCL,SHCL     00060
      READ 10,NV,NA,NAT,MACT,NSH,NSHT,MSHT,NSU,NSUT,MSUT,MINTOT         00070
      REWIND 2                                                          00080
      READ TAPE 2, NF,NACR,NACC,NSHR,NSHC,NSUR,NSUC,XREF,YREF,(FXPTS(I),00090
     1I=1,NF),(FYPTS(I),I=1,NF),((ACLDL(I,J),I=1,NACR),J=1,NACC),((SHLDL00100
     2(I,J),I=1,NSHR),J=1,NSHC),((SULDL(I,J),I=1,NSUR),J=1,NSUC)        00110
      READ TAPE 2,XMEANX,YMEANY,NMAX,MMAX,PHI                           00120
      READ TAPE 2,XMEANX,YMEANY,NMAX1,MMAX1,BOT                         00130
C     NV=TOTAL NO. OF VEHICLES ON ARRIVAL LIST, NA=TOTAL NO. OF         00140
C     AIRCRAFT,NAT=TOTAL NUMBER OF AIRCRAFT TYPES,NACR=NO. OF ROWS      00150
C     IN ACLDL,NACC=NO. OF COLUMNS IN ACLDL,MACT=TOTAL NO. OF           00160
C     AIRCRAFT MINE TYPES,NSH=TOTAL NO. OF SHIPS,NSHT=TOTAL NO. OF      00170
C     SHIP TYPES,NSHR=NO. OF ROWS IN SHLDL,NSHC=NO. OF COLUMNS IN       00180
C     SHLDL,MSHT=TOTAL NO. OF SHIP MINE TYPES,NSU=TOTAL NUMBER OF SUBMA-00190
C     RINES, NSUT=TOTAL NO. OF SUBMARINE TYPES,NSUR=NO. OF ROWS IN SULDL,00200
C     NSUC=NO. OF COLUMNS IN SULDL,MSUT=TOTAL NO. OF SUBMARINE MINE TYPES,00210
C     MINTOT=TOTAL NO. OF MINES.                                        00220
   10 FORMAT(11I5)                                                      00230
      READ 1,((VAL(I,J),J=1,11),I=1,NV)                                 00240
    1 FORMAT(F8.1,4F3.0,6F10.0)                                         00250
      READ 2,((ACCL(I,J),J=1,6),I=1,NAT)                                00260
    2 FORMAT(4F5.0,2F4.2)                                               00270
      READ 3,((ACMCL(I,J),J=1,14),I=1,MACT)                             00280
    3 FORMAT(F4.0,F6.2,2F6.0,5F6.2,2F3.0,3F6.2)                         00290
      READ 4,((BR(I,J),J=1,6),I=1,NA)                                   00300
    4 FORMAT(6F10.0)                                                    00310
      READ 65,((MDL(I,J),J=1,8),I=1,MINTOT)                             00320
   65 FORMAT(4I5,I8,3I5)                                                00330
      READ 303,((SHCL(I,J),J=1,9),I=1,NSHT)                             00340
  303 FORMAT (9F5.0)                                                    00350
      READ 302,((SHMCL(I,J),J=1,8),I=1,MSHT)                            00360
```

```
  302 FORMAT(8F5.0)                                                              00370
      READ 409,((SUCL(I,J),J=1,17),I=1,NSUT)                                     00380
  409 FORMAT (F2.0,5F5.0,F3.2,F4.1,3F3.2,3F4.1,2F3.2,F10.0))                     00390
      READ 411,TAM                                                               00400
  411 FORMAT(F10.1)                                                              00410
      READ 430, ((SUMCL(I,J),J=1,13),I=1,MSUT)                                   00420
  430 FORMAT(F5.0,2F5.2,F3.0,F5.0,6F5.2,F5.0,F5.2)                               00430
      READ 8001,IRMAT                                                            00440
 8001 FORMAT(8I10)                                                               00440
      READ 8002, NREP                                                            00460
 8002 FORMAT(I3)                                                                 00470
      REWIND 3                                                                   00480
      IF(TAM) 667,667,666                                                        00490
  667 DO 668 ISUB = 1,NV                                                         00500
      IF(VAL(ISUB,2)-9.)668,668,669                                              00510
  669 TAM = VAL(ISUB,1)                                                          00520
      GO TO 666                                                                  00530
  668 CONTINUE                                                                   00540
  666 WRITE TAPE 3,XREF,YREF,EXPTS,EYPTS,ACLDL,SHLDL,SULDL,VAL,ACCL,SHCL         00550
     1,SUCL,ACMCL,SHMCL,SUMCL,BR,MDL,NV,NA,NAT,NACR,NACC,MACT,NSH,NSHT,N         00560
     2SHR,NSHC,MSHT,NSU,NSUT,NSUR,NSUC,MSUT,MINTOT,NF,TAM                       00570
      DO 8500 IREP=1,NREP                                                        00650
      IR = IRMAT(IREP)                                                           00660
      PRINT 8003,IREP                                                            00670
 8003 FORMAT(1H1,8H RUN NO.,I3//)                                                00680
      PRINT 40                                                                   00690
   40 FORMAT(//51X,19H  SUMMARY NARRATIVE///2X,5H TIME,10X,6H EVENT//)           00700
C DUPLICATES BOMBRACK MATRIX FOR USE IN ITERATIVE RUNS.                          00710
      DO 8010 LL1=1,NV                                                           00720
```

```
      DO 8020 LL2=1,6                                                    00730
 8020 BRTEMP(LL1,LL2) = BR(LL1,LL2)                                      00740
 8010 CONTINUE                                                           00745
      DO 8055 III =1,30                                                  00750
 8055 OUTPUT(III)=0.                                                     00760
      OUTPUT(28)=IR                                                      00770
      OUTPUT(29)=IREP                                                    00780
      WRITE TAPE 3,OUTPUT                                                00790
C  SELECTION OF VEHICLES IN ORDER OF APPEARANCE ON VEHICLE ARRIVAL LIST  00800
      DO9999 IABC=1,NV                                                   00810
      I=IABC                                                             00820
C  DETERMINATION AND INDEXING OF VEHICLE TYPE                          00830
      IF(VAL(I,2)-100.)305,102,102                                       00840
  305 IF(VAL(I,2)-10.) 405,306,306                                       00850
  405 J = VAL(I,2)                                                       00860
      S=SUCL(J,3)*2025.37/ 60.                                           00870
      TD100=SUCL(J,17)/(SUCL(J,2)*2025.37/ 60.)                          00880
      MR=0                                                               00890
      BNE=SUCL(J,4)                                                      00900
      ANE=SUCL(J,5)                                                      00910
      GNE=SUCL(J,6)                                                      00920
      JJ = 3                                                             00930
      GO TO 103                                                          00940
  306 J=VAL(I,2)-9.                                                      00950
      S=SHCL(J,3)*2025.37/60.                                            00960
      BNE=SHCL(J,4)                                                      00970
      ANE=SHCL(J,5)                                                      00980
      GNE=SHCL(J,6)                                                      00990
      JJ=2                                                               01000
      IDAM=0                                                             01010
      GO TO 103                                                          01020
  102 J = VAL(I,2) -99.                                                  01030
      S = ACCL(J,3) * 2025.37 / 60.                                      01040
      JJ = 1                                                             01050
  103 MM = 2                                                             01060
      TD(1) = 0.                                                         01070
```

```
      NTOTLG = VAL(I,4)                                                  01080
      DO301 IL=1,6                                                       01090
      NIL = IL+ 5                                                        01100
  301 XLCOD(IL) = VAL(I,NIL)                                             01110
      DO 104 KK = 1,NTOTLG                                               01120
C DETERMINATION OF LEGS ASSIGNED VEHICLE AND TIME INCREMENTS IN FIELD    01130
      CALL XTLEGN(XLCOD,L)                                               01140
      GO TO(105,106,107),JJ                                              01150
  105 N1 = 3.*ACLDL(L,2)+7.                                              01160
C DETERMINES DELTA TIME BETWEEN LEGS IF MORE THAN ONE LEG ASSIGNED       01170
      DO108 KKK = 1,N1                                                   01180
  108 XLDL(KKK) = ACLDL(L,KKK)                                           01190
      N = ACLDL(L,2)+1.                                                  01200
      GO TO 308                                                          01210
  107 N = SULDL(L,2) + 1.                                                01220
      DO 406 JIK = 1,4                                                   01230
  406 XLDL(JIK) = SULDL(L,JIK)                                           01240
      DO 407 JI1 = 1,N                                                   01250
      NJ1=JI1 * 5-1                                                      01260
      NJ11 = NJ1 - (JI1 - 1)* 2                                          01270
      DO 408 JI2= 1,3                                                    01280
      NJ2=NJ1 + JI2                                                      01290
      NJ22 = NJ11 + JI2                                                  01300
  408 XLDL(NJ22) = SULDL(L,NJ2)                                          01310
  407 CONTINUE                                                           01320
      GO TO 308                                                          01330
  106 N1=3.*SHLDL(L,2)+7.                                                01340
      DO 307 KKK=1,N1                                                    01350
  307 XLDL(KKK)=SHLDL(L,KKK)                                             01360
      N=SHLDL(L,2)+1.                                                    01370
  308 IF(KK-1)109,109,110                                                01380
  110 XIPX = XLDL(3)                                                     01390
      YIPY = XLDL(4)                                                     01400
      DX = (XIPX - TPX)**2                                               01410
      DY = (YIPY - TPY)**2                                               01420
      TE = (SQRTF(DX+DY))/S                                              01430
```

```
      TD(MM) = TE + TD(MM-1)                                                01440
      MM = MM + 1                                                           01450
C  DETERMINES DELTA TIME BETWEEN POINTS ON A LEG                           01460
  109 DO111 II = 1,N                                                        01470
      I1 = 3*(II-1)+6                                                       01480
      I2 = 3*(II-2)+6                                                       01490
      I3 = I1+1                                                             01500
      I4 = I2+1                                                             01510
      IF(II-1)112,112,113                                                   01520
  112 DX = (XLDL(6) - XLDL(3))**2                                           01530
      DY = (XLDL(7) - XLDL(4))**2                                           01540
      GO TO 114                                                             01550
  113 DX = (XLDL(I1)-XLDL(I2))**2                                           01560
      DY = (XLDL(I3)-XLDL(I4))**2                                           01570
  114 TE = (SQRTF(DX+DY))/S                                                 01580
      TD(MM) = TE + TD(MM-1)                                                01590
      MM = MM + 1                                                           01600
      TPX = XLDL(I1)                                                        01610
  111 TPY = XLDL(I3)                                                        01620
  104 CONTINUE                                                              01630
   27 GO TO (28,310,415),JJ                                                 01640
C DETERMINE IF SUBMARINE IS DETECTED AND TIME OF DETECTION.                 01650
  415 PROB(1)=(1.0-SUCL(J,9))*(1.0- SUCL(J,11))                             01660
      PROB(2) = PROB(1) + (1.0-SUCL(J,9))*SUCL(J,11)*(1.0 - SUCL(J,12))     01670
      PROB(3) = PROB(2) + (1.0-SUCL(J,9))*SUCL(J,11)* SUCL(J,12)*(1.        01680
     10 - SUCL(J,13))                                                       01690
      PROB(4) = PROB(3) + (1.0-SUCL(J,9))* SUCL(J,11)*SUCL(J,12)* SU        01700
     1CL(J,13)                                                              01710
      PROB(5) = PROB(4) + SUCL(J,9) * (1.0 - SUCL(J,10)) * (1.0 - SUCL(J    01720
     1,15))                                                                 01730
      PROB(6) = PROB(5) + SUCL(J,9) * (1.0- SUCL(J,10)) * SUCL(J,15)*       01740
     1(1.0 - SUCL(J,16))* (1.0- SUCL(J,14))                                 01750
      PROB(7) = PROB(6) + SUCL(J,9) * (1.0- SUCL(J,10)) * SUCL(J,15)*       01760
     1(1.0- SUCL(J,16))*SUCL(J,14)                                          01770
      PROB(8)= PROB(7) + SUCL (J,9) * (1.0-SUCL(J,10))* SUCL(J,15) *        01780
     1SUCL(J,16)                                                            01790
```

```
      PROB(9) = PROB(8) + SUCL(J,9)* SUCL(J,10)*SUCL(J,12)*(1.0-S          01800
     1UCL(J,13))                                                           01810
      PROB(10) = PROB(9) + SUCL(J,9) * SUCL(J,10) * SUCL(J,12) * SUCL(J    01820
     1,13)                                                                 01830
      PROB(11) = PROB(10) + SUCL(J,9) * SUCL(J,10) * (1.0- SUCL(J,12))     01840
     1*(1.0- SUCL(J,15))                                                   01850
      PROB(12) = PROB(11) + SUCL(J,9) * SUCL(J,10)*(1.0- SUCL(J,12))       01860
     1*SUCL(J,15)* SUCL(J,16)                                              01870
      PROB (13) = PROB(12) +SUCL(J,9) *SUCL(J,10)*(1.0- SUCL(J,12))        01880
     1* SUCL(J,15)*(1.0- SUCL(J,16))* (1.0-SUCL(J,14))                     01890
      PROB(14)=PROB(13) + SUCL(J,9)*SUCL(J,10)*(1.0-SUCL(J,12))            01900
     1* SUCL(J,15)* (1.0- SUCL(J,16))* SUCL(J,14)                          01910
      CALL RAN1(IR,RN)                                                     01920
C PSEUDO DETECTION TIME,TYPE OF DETECTION AND/OR ATTRITION,IF ANY,DETERMINE01930
C DEVELOPMENT OF SUBMARINE NARRATIVE.                                      01940
      DT= TD(MM-1)*RN                                                      01950
      TOTTIM = TD(MM-1)                                                    01960
      MQ = MM-1                                                            01965
      CALL TIMDIF(VAL(I,1),TAM,TDIF)                                       01970
      GO TO 31                                                             01980
C DETERMINE IF SHIP SUFFERS DAMAGE AND TIME OF DAMAGE                      01990
  310 PU = SHCL(J,9)                                                       02000
      ALPHA = LOGF(PU+1.)/.7                                               02010
      T = .5*(LOGF(2.1/LOGF(PU+1.))                                        02020
      IF(T-TD(MM-1))311,311,312                                            02030
  311 CALL RAN1(IR,RN)                                                     02040
      DT = TD(MM-1)*RN                                                     02050
C     PRINT 315,TD(MM-1),T,DT                                              02060
      GO TO 31                                                             02070
  312 PDAM = EXPF(1.4*ALPHA*TD(MM-1))-1.                                   02080
      CALL RAN1(IR,RN)                                                     02090
      IF(PDAM-RN)313,311,311                                               02100
C SET DUMMY DAMAGE TIME GREATER THAN TOTAL TIME IN FIELD                   02110
  313 DT = TD(MM-1) + 60.                                                  02120
C     PRINT 315,TD(MM-1),T,DT                                              02130
C 315 FORMAT(27H DAMAGE TEST. TOTAL TIME = ,F10.1, 5H T = ,F5.1, 6H DT =   02140
```

```
C     1 ,F10.1/)                                                        02150
      GO TO 31                                                          02160
C  DETERMINE IF AIRCRAFT SURVIVES                                      02180
   28 CALL RAN1(IR,RN)                                                  02190
      IF(RN-ACCL(J,6))29,29,30                                          02200
C  SET DUMMY DEAD TIME GREATER THAN TOTAL TIME IN FIELD                 02210
   29 DT = TD(MM-1)+5.                                                  02220
C     PRINT 32,VAL(I,3),DT                                             02230
C  32 FORMAT(8H A/C NO.,F4.0,29H SURVIVES. DUMMY DEAD TIME = ,F5.1//)   02240
      GO TO 31                                                          02250
C  DETERMINE DEAD TIME                                                  02260
   30 CALL RAN1(IR,RN)                                                  02270
      DT = TD(MM-1)*RN                                                  02280
C     PRINT 33,VAL(I,3),DT                                             02290
C  33 FORMAT(8H A/C NO.,F4.0,19H DIES. DEAD TIME = ,F5.1//)             02300
   31 MM = 2                                                            02310
      DO480 IL=1,6                                                      02320
      NIL = IL+ 5                                                       02330
  480 XLCOD(IL) = VAL(I,NIL)                                            02340
      GO TO(316,317,420),JJ                                             02350
C  BEGIN DEVELOPEMENT OF SUBMARINE NARRATIVE                            02360
  420 CALL XTLEGN(XLCOD,L)                                              02370
      DO 500 K=1,NTOTLG                                                 02380
      NMINES = SULDL(4,2)+1.                                            02390
      DO 501 NMIN = 1, NMINES                                           02400
      DO 502 III=1,30                                                   02410
  502 OUTPUT(III) = 0.                                                  02420
      IF(K-1)503,503,504                                                02430
  503 IF(NMIN-1)505,505,504                                             02440
  505 PRINT 506,VAL(I,1),VAL(I,3),L                                     02450
  506 FORMAT(F10.1,14H SUBMARINE NO.,F4.0,26H ARRIVES AT IP FOR LEG NO.,  02460
     1I4 /)                                                             02470
      GO TO 507                                                         02480
  504 IF(NMIN - NMINES)507,508,508                                      02490
C SUBMARINE MINE FIRING DELAY ROUTINE.                                  02500
  507 SDELAY = 0.                                                       02510
```

```
      LDFLAG=1                                                          02520
      TREF=TD(MM)                                                       02530
      NM = 5*(NMIN-1)+5                                                 02540
      M1 = SULDL(L,NM)                                                  02550
  513 CALL RAN1(IR,RN)                                                  02560
      IF(SUCL(J,7)-RN)510,509,509                                       02570
  510 CALL RAN1(IR,DELAY)                                               02580
      SDELAY = SDELAY + DELAY                                           02590
      IF(SDELAY-SUCL(J,8))511,511,512                                   02600
  511 TREF = SDELAY + TD(MM)                                            02610
      IF(TREF - TD(MM+1))513,512,512                                    02620
C STORE MINES NOT FIRED DUE TO EXCESSIVE FIRING DELAY.                  02630
  512 MR = MR+1                                                         02640
      SMROB(MR)=M1                                                      02650
      CALL TIMCON(VAL(I,1),TREF,TLAY)                                   02660
      PRINT 514,TLAY,SULDL(L,NM)                                        02670
  514 FORMAT(F10.1,9H MINE NO.,F4.0,38H NOT LAYED DUE TO EXCESSIVE TIME 02680
     1DELAY /)                                                          02690
      MLFLAG = 0                                                        02700
      GO TO 520                                                         02710
  509 IF(TREF-DT)516,515,515                                            02720
C DETERMINE TYPE OF EVASIVE ACTION/DETECTION/ATTRITION EVENT.           02730
  515 CALL RAN1(IR,RN)                                                  02740
      DO 517 JCIG = 1,14                                                02750
      IDET = JCIG                                                       02760
      IF(PROB(JCIG)-RN)517,518,518                                      02770
  517 CONTINUE                                                          02780
  518 CALL TIMCON(VAL(I,1),DT,TDT)                                      02790
      GO TO (539,539,521,522,523,524,525,526,527,528,529,530,531,532),ID 02800
     1ET                                                                02810
  539 DT = 2.0*TOTTIM                                                   02813
      GO TO 516                                                         02816
  521 PRINT 533,TDT,VAL(I,3)                                            02820
      IDET1=1                                                           02830
      GO TO 550                                                         02840
  522 PRINT 533,TDT,VAL(I,3)                                            02850
```

```
533 FORMAT(F10.1,8H SUB NO.,F4.0,54H SUSPENDS OPS DUE TO ERRONEOUS KNO   02860
   1WLDEGE OF DETECTION. /)                                             02870
    IDET1=1                                                             02880
    GO TO 560                                                           02890
523 PRINT 534,TDT,VAL(I,3)                                              02900
534 FORMAT(F10.1,8H SUB NO.,F4.0,10H DETECTED. /)                       02910
    GO TO 516                                                           02920
524 PRINT 535,TDT,VAL(I,3)                                              02930
535 FORMAT(F10.1,8H SUB NO.,F4.0,23H DETECTED AND ATTACKED. /)          02940
    IDET1=1                                                             02950
    GO TO 550                                                           02960
525 PRINT 535,TDT,VAL(I,3)                                              02970
    IDET1=2                                                             02980
    GO TO 560                                                           02990
526 PRINT 535,TDT,VAL(I,3)                                              03000
    IDET1=2                                                             03010
    GO TO 550                                                           03020
527 PRINT 536,TDT,VAL(I,3)                                              03030
536 FORMAT(F10.1,8H SUB NO.,F4.0,27H DETECTED AND SUSPENDS OPS. /)       03040
    IDET1=1                                                             03050
    GO TO 550                                                           03060
528 PRINT 536,TDT,VAL(I,3)                                              03070
    IDET1=1                                                             03080
    GO TO 560                                                           03090
529 PRINT 534,TDT,VAL(I,3)                                              03100
    GO TO 516                                                           03110
530 PRINT 535,TDT,VAL(I,3)                                              03120
    IDET1=2                                                             03130
    GO TO 550                                                           03140
531 PRINT 535,TDT,VAL(I,3)                                              03150
    IDET1=1                                                             03160
    GO TO 550                                                           03170
532 PRINT 535,TDT,VAL(I,3)                                              03180
    IDET1=2                                                             03190
560 GO TO(540,544),IDET1                                                03200
540 CALL RAN1(IR,RN)                                                    03210
```

```
      TRET=DT+2.*TD100+(TDIF-DT)*RN                                     03220
  545 CALL TIMCON(VAL(I,1),TRET,TTRET)                                  03230
      IDET1=3                                                           03240
      IF(TAM - TTRET)550,550,541                                        03250
  541 GO TO(537,591,591),LDFLAG                                         03260
  591 IF(K - NTOTLG) 537,592,592                                        03270
  592 PRINT 593                                                         03280
  593 FORMAT(22X,32H MISSION COMPLETED. LEAVES AREA. //)                 03290
      GO TO 9999                                                        03300
  537 PRINT 542,TTRET,VAL(I,3)                                          03310
  542 FORMAT(F10.1,8H SUB NO.,F4.0,13H RESUMES OPS. /)                  03320
  594 DO 543 JUP =MM,134                                                03330
  543 TD(JUP+1) = TRET + (TD(JUP+1)-TD(MM))+10.*RN                      03340
      TOTTIM = TD(MQ)                                                   03345
      TREM = TOTTIM -TD(MM)                                             03350
      TD(MM) = TRET +10.*RN                                             03360
      CALL RAN1(IR,RN)                                                  03370
      DT = TRET + TREM*RN                                               03380
      GO TO(507,595,589),LDFLAG                                         03390
  595 MM = MM + 1                                                       03400
      GO TO 589                                                         03410
  544 CALL RAN1(IR,RN)                                                  03420
      TRET = DT + 2.*TD100 +2.*(TDIF-DT)*RN                             03430
      GO TO 545                                                         03440
  550 GO TO(546,547,548),IDET1                                          03450
  546 PRINT 549                                                         03460
  549 FORMAT(22X,45H ABORTS MISSION. MINES REMAINING ON BOARD ARE )     03470
      GO TO 561                                                         03480
  547 PRINT 551                                                         03490
  551 FORMAT(22X,40H ATTRITTED. MINES REMAINING ON BOARD ARE )          03500
      GO TO 561                                                         03510
  548 TQ = TDIF - TD100                                                 03520
      CALL TIMCON(VAL(I,1),TQ,TDT)                                      03530
      PRINT 552,TDT,VAL(I,3)                                            03540
  552 FORMAT(F10.1,8H SUB NO.,F4.0,68H ABORTS MISSION DUE TO TIME LIMITA 03550
     1TION. MINES REMAINING ON BOARD ARE )                              03560
```

```
  561 DO 556 IDOT = 1,MR                                                    03570
      PRINT 538,SMROB(IDOT)                                                 03580
  556 CONTINUE                                                              03590
  553 DO 554 NLEGR = K,NTOTLG                                               03600
      NMINES =-SULDL(L,2)                                                   03610
      DO 555 NMINR = NMIN,NMINES                                            03620
      NM = 5*(NMINR-1)+5                                                    03630
      PRINT 538,SULDL(L,NM)                                                 03640
  538 FORMAT(90X,F4.0)                                                      03650
  555 CONTINUE                                                              03660
      NMIN=1                                                                03670
      CALL XTLEGN(XLCOD,L)                                                  03680
  554 CONTINUE                                                              03690
      GO TO 9999                                                            03700
C DETERMINE SUBMARINE FIRING POSITION.                                      03710
  516 GO TO(519,585,589),LDFLAG                                             03720
  519 IF(TREF-TDIF)563,562,562                                              03730
  562 PRINT 552,TAM,VAL(I,3)                                                03740
      GO TO 561                                                             03750
  563 MLFLAG = 1                                                            03760
      XDELE = 0.                                                            03770
      YDELE = 0.                                                            03775
      CALL POSIT(IR,BNE,ANE,GNE,SULDL(L,NM+1),SULDL(L,NM+2),XDELE,YDELE,    03780
     1XPOSIT,YPOSIT)                                                        03790
      CALL TIMCON(VAL(I,1),TREF,TLAY)                                       03800
      PRINT 265,TLAY,SULDL(L,NM)                                            03810
      MT2=NOL(M1,2)                                                         03820
      MFLA =SUMCL(MT2,4)                                                    03830
C DETERMINE IF MINE IS SELF-PROPELLED AND TERMINAL POSIT OF SELF-PROPELLED  03840
      IF(SUMCL(MT2,8))576,576,567                                           03850
  567 CALL RAN1(IR,RN)                                                      03860
      IF(SUMCL(MT2,8)-RN)569,570,570                                        03870
  569 DX=XPOSIT-SULDL(L,NM+3)                                               03880
      DY=YPOSIT-SULDL(L,NM+4)                                               03890
      HYP=SQRTF(DX**2+DY**2)                                                03900
      CALL RAN1(IR,RN)                                                      03910
      RUN=HYP*RN                                                            03920
```

```
C DETERMINE IF RUN IS NORMAL FOR SELF-PROPELLED MINE. IF NOT, DETERMINE TYP03930
C FAILURE AND CORRESPONDING MINE POSIT.                                 03940
      CALL RAN1(IR,RN)                                                  03950
      IF(SUMCL(MT2,9)-RN)571,572,572                                    03960
  571 IF(SUMCL(MT2,10)-RN)573,574,574                                   03970
  572 XPOSIT=XPOSIT+RUN*DX/HYP                                          03980
      YPOSIT=YPOSIT+RUN*DY/HYP                                          03990
      OUTPUT(13)=1.                                                     04000
      IF(RUN-500.)575,575,576                                           04010
  575 OUTPUT(12)=1.                                                     04020
      GO TO 576                                                         04030
  573 CALL RAN1(IR,RN)                                                  04040
      OUTPUT(15)=1.                                                     04050
      DRUN=(HYP-RUN)*RN+RUN                                             04060
      XPOSIT=XPOSIT+DRUN*DX/HYP                                         04070
      YPOSIT=YPOSIT+DRUN*DY/HYP                                         04080
      GO TO 576                                                         04090
  574 OUTPUT(14)=1.                                                     04100
      CALL RAN1(IR,RN)                                                  04110
      GYRUN=(HYP-RUN)*RN                                                04120
      CALL RAN1(IR,RN)                                                  04130
      GYPHI=6.28318*RN                                                  04140
      XPOSIT=XPOSIT+RUN*DX/HYP+GYRUN*COSF(GYPHI)                        04150
      YPOSIT=YPOSIT+RUN*DY/HYP+GYRUN*SINF(GYPHI)                        04160
      GO TO 576                                                         04170
  570 SIGMA=SUMCL(MT2,12)/1.1744                                        04180
      CALL RNORM(SIGMA,XE,IR)                                           04190
      CALL RNORM(SIGMA,YE,IR)                                           04200
      XPOSIT=SULDL(L,NM+3)+XE                                           04210
      YPOSIT=SULDL(L,NM+4)+YE                                           04220
  576 CALL DEPTH(XPOSIT,YPOSIT,PHI,NMAX,MMAX,YMEANY,XMEANX,WD)          04230
      IF(WD)577,577,578                                                 04240
  577 OUTPUT(9)=1.                                                      04250
      CALL TAPO(OUTPUT,TLAY,MT2,MDL(M1,3),MDL(M1,8),MDL(M1,7),MDL(M1,5),04252
     1MDL(M1,6),M1,XPOSIT,YPOSIT,WD,MFLAG)                              04254
      GO TO 520                                                         04260
```

```
  578 CALL TAPO(OUTPUT,TLAY,MT2,MDL(M1,3),MDL(M1,8),MDL(M1,7),MDL(M1,5),    04270
     1MDL(M1,6),M1,XPOSIT,YPOSIT,WD,MFLAG)                                 04280
      CALL DEPTH(XPOSIT,YPOSIT,BOT,NMAX1,MMAX1,YMEANY,XMEANX,BTMTYP)       04290
      CALL RAN1(IR,RN)                                                     04300
      PD=(SUMCL(MT2,2)/5.)*BTMTYP                                          04310
      IF(PD-RN)579,580,580                                                 04320
  580 OUTPUT(3)=1.                                                         04330
  579 GO TO (581,581,582,582,581,581),MFLAG                                04340
  581 CALL MOOR(OUTPUT,IR,MFLAG,WD,MDL(M1,4),SUMCL(MT2,6),SUMCL(MT2,7),S   04350
     1UMCL(MT2,13))                                                        04360
      GO TO 520                                                            04370
  582 CALL BOTTOM(OUTPUT,IR,MFLAG,BTMTYP,SUMCL(MT2,3),SUMCL(MT2,5),SUMCL   04380
     1(MT2,13))                                                            04390
  520 MM=MM+1                                                              04400
      IF(MLFLAG)501,501,583                                                04410
  583 WRITE TAPE 3, OUTPUT                                                 04420
  501 CONTINUE                                                             04430
  508 IF(DT-TD(MM))584,584,585                                             04440
  584 LDFLAG=2                                                             04450
      GO TO 515                                                            04460
  585 CALL TIMCON(VAL(I,1),TD(MM),TTP)                                     04470
      IF(TTP-TAM)599,596,596                                               04480
  599 MM=MM+1                                                              04490
      PRINT 586,TTP,VAL(I,3),L                                             04500
  586 FORMAT(F10.1,8H SUB NO.,F4.0,26H ARRIVES AT TP FOR LEG NO.,I4 //)    04510
      IF(K - NTOTLG) 587,9999,9999                                         04520
C DETERMINE IF SUBMARINE SURVIVES BETWEEN TP AND IP FOR NEXT LEG.          04530
  587 IF(DT-TD(MM))588,588,589                                             04540
  588 LDFLAG=3                                                             04550
      GO TO 515                                                            04560
  589 CALL TIMCON(VAL(I,1),TD(MM),TIP)                                     04570
      IF(TIP-TAM)597,596,596                                               04580
  596 PRINT 552,TAM,VAL(I,3)                                               04590
      GO TO 561                                                            04600
  597 MM=MM+1                                                              04610
      CALL XTLEGN(XLCOD,L)                                                 04620
```

```
      PRINT 598,TIP,VAL(I,3),L                                          04630
  598 FORMAT(F10.1,8H SUB NO.,F4.0,26H ARRIVES AT IP FOR LEG NO.,I4/)    04640
  500 CONTINUE                                                          04650
      GO TO 9999                                                        04660
C BEGIN DEVELOPEMENT OF SHIP NARRATIVE                                  04670
  317 CALL XTLEGN(XLCOD,L)                                              04680
      DO 318 K = 1,NTOTLG                                               04690
      NMINES = SHLDL(L,2)+1.                                            04700
      DO319 NMIN = 1,NMINES                                             04710
      DO 320  III=1,30                                                  04720
  320 OUTPUT(III)=0.                                                    04730
      IF(K-1)321,321,322                                                04740
  321 IF(NMIN-1)323,323,322                                             04750
  323 PRINT 324,VAL(I,1),VAL(I,3),L                                     04760
  324 FORMAT(F10.1, 9H SHIP NO.,F4.0,26H ARRIVES AT IP FOR LEG NO.,I4/) 04770
      GO TO 325                                                         04780
  322 IF(NMIN-NMINES)325,452,452                                        04790
C DETERMINE IF MINE RELEASES ON FIRST ATTEMPT                           04800
  325 CALL RAN1(IR,RN)                                                  04805
      IF(SHCL(J,7)-RN)326,327,327                                       04810
  327 DELAY = 0.                                                        04820
      GO TO 328                                                         04830
  326 CALL RAN1(IR,RN)                                                  04840
      DELAY = SHCL(J,8)*RN                                              04850
  328 TDUM = TD(MM)+DELAY                                               04860
      IF(IDAM)329,329,347                                               04870
  329 LDFLAG = 1                                                        04880
      IF(TDUM-DT)347,331,331                                            04890
  331 IDAM=1                                                            04900
      CALL TIMCON(VAL(I,1),DT,TDT)                                      04910
C DETERMINE IF SHIP IS DAMAGED AND TYPE OF DAMAGE.                      04920
      CALL RAN1(IR,RN)                                                  04930
      NDAM = RN*10.+1.                                                  04940
      GO TO(340,340,341,341,342,342,343,343,344,344,344),NDAM           04950
  340 PRINT 345,TDT,VAL(I,3)                                            04960
  345 FORMAT(F10.1, 9H SHIP NO.,F4.0,37H DAMAGED. SPEED REDUCED BY ONE-H 04970
     1ALF. /)                                                           04980
```

```
      TREF2 = TD(MM)                                                    04990
      TREF1=(TD(MM)-DT)*2.+DT                                           05000
      DO 346 JU = MM,134                                                05010
  346 TD(JU+1)=(TD(JU+1)-TREF2)*2.+TREF1                                05020
      TDUM=TREF1 + DELAY                                                05030
      S = S/2.                                                          05040
      GO TO(347,460,470),LDFLAG                                         05050
  341 PRINT 348,TDT,VAL(I,3)                                            05060
  348 FORMAT(F10.1,9H SHIP NO.,F4.0,37H DAMAGED. NAVIGATIONAL ERROR DOUB 05070
     1LED./)                                                            05080
      BNE = BNE*2.                                                      05090
      ANE=ANE*2.                                                        05100
      GNE=GNE*2.                                                        05110
      GO TO(347,460,470),LDFLAG                                         05120
  342 PRINT 349,TDT,VAL(I,3)                                            05130
  349 FORMAT(F10.1,9H SHIP NO.,F4.0,31H DAMAGED. NO EFFECT ON MISSION./) 05140
      GO TO(347,460,470),LDFLAG                                         05150
  343 PRINT 350,TDT,VAL(I,3)                                            05160
  350 FORMAT(F10.1,9H SHIP NO.,F4.0,35H SUNK. MINES REMAINING ON BOARD A 05170
     1RE )                                                              05180
      GO TO 351                                                         05190
  344 PRINT 352,TDT,VAL(I,3)                                            05200
  352 FORMAT(F10.1,9H SHIP NO.,F4.0,54H DAMAGED. ABORTS MISSION. MINES R 05210
     1EMAINING ON BOARD ARE )                                           05220
  351 DO 353 NLEGR = K,NTOTLG                                           05230
      NMINES = SHLDL(L,2)                                               05240
      DO354 NMINR = NMIN,NMINES                                         05250
      NM = 3*(NMINR-1)+5                                                05260
      PRINT 355,SHLDL(L,NM)                                             05270
  355 FORMAT(77X,F4.0)                                                  05280
  354 CONTINUE                                                          05290
      NMIN = 1                                                          05300
      CALL XTLEGN(XLCOD,L)                                              05310
  353 CONTINUE                                                          05320
      GO TO 9999                                                        05330
  347 NM = 3*(NMIN-1)+5                                                 05340
```

```
      M1 = SHLDL(L,NM)                                                  05350
      CALL TIMCON(VAL(I,1),TDUM,TLAY)                                   05360
      PRINT 265,TLAY,SHLDL(L,NM)                                        05370
      MT2 = MDL(M1,2)                                                   05380
      IF(DELAY)357,357,356                                              05390
C DETERMINE SHIP DELIVERED MINE POSIT.                                  05400
  356 DX = SHLDL(L,NM+4)-SHLDL(L,NM+1)                                  05410
      DY = SHLDL(L,NM+5)-SHLDL(L,NM+2)                                  05420
      HYP = SQRTF(DX**2 + DY**2)                                        05430
      XDELE = DELAY*S*(DX/HYP)                                          05440
      YDELE = DELAY*S*(DY/HYP)                                          05450
C     PRINT 2001,XDELE,YDELE                                            05460
      GO TO 358                                                         05470
  357 YDELE = 0.                                                        05480
      XDELE = 0.                                                        05490
C     PRINT 2001,XDELE,YDELE                                            05500
C2001 FORMAT(/9H XDELE = ,F10.3,9H YDELE = ,F10.3 /)                    05510
  358 CALL POSIT(IR,BNE,ANE,GNE,SHLDL(L,NM+1),SHLDL(L,NM+2),XDELE,YDELE,  05520
     1XPOSIT,YPOSIT)                                                    05530
      CALL DEPTH(XPOSIT,YPOSIT,PHI,NMAX,MMAX,YMEANY,XMEANX,WD)          05540
C     PRINT 2003,WD                                                     05550
C2003 FORMAT(/6H WD = ,F10.3 /)                                         05560
      CALL DEPTH(XPOSIT,YPOSIT,BOT,NMAX1,MMAX1,YMEANY,XMEANX,BTMTYP)    05570
C     PRINT 2004,BTMTYP                                                 05580
C2004 FORMAT(/10H BTMTYP = ,F10.3 /)                                    05590
      MFLAG = SHMCL(MT2,4)                                              05600
      CALL TAPO(OUTPUT,TLAY,MT2,MDL(M1,3),MDL(M1,8),MDL(M1,7),MDL(M1,5),  05610
     1MDL(M1,6),M1,XPOSIT,YPOSIT,WD,MFLAG)                              05620
      IF(WD-20.)362,362,363                                             05630
  362 CALL RAN1(IR,RN)                                                  05640
      PDHB = SHMCL(MT2,2)                                               05650
      PD=(PDHB/5.)*BTMTYP                                               05660
      IF(PD-RN)363,364,364                                              05670
  364 OUTPUT(3)=1.                                                      05680
  363 GO TO(365,365,367,367,365,365),MFLAG                              05690
  365 CALL MOOR(OUTPUT,IR,MFLAG,WD,MDL(M1,4),SHMCL(MT2,6),SHMCL(MT2,7),  05700
```

```
     1SHMCL(MT2,8))                                                          05710
      GO TO 442                                                              05720
  367 CALL BOTTOM (OUTPUT,IR,MFLAG,BTMTYP,SHMCL(MT2,3),SHMCL(MT2,5),SHMC     05730
     1L(MT2,8))                                                              05740
  442 MM=MM+1                                                                05750
      WRITE TAPE 3,OUTPUT                                                    05760
  319 CONTINUE                                                               05770
C  DETERMINE IF SHIP SURVIVES TRANSIT FROM TP TO IP FOR NEXT LEG             05780
  452 IF(IDAM)453,453,460                                                    05790
  453 IF(DT-TD(MM))455,455,460                                               05800
  455 LDFLAG=2                                                               05810
      DELAY=0.                                                               05820
      GO TO 331                                                              05830
  460 CALL TIMCON(VAL(I,1),TD(MM),TTP)                                       05840
      MM=MM+1                                                                05850
      PRINT 461,TTP,VAL(I,3),L                                               05860
  461 FORMAT(F10.1,9H SHIP NO.,F4.0,26H ARRIVES AT TP FOR LEG NO.,I4 //)     05870
      IF(K - NTOTLG) 462,9999,9999                                           05880
  462 IF(IDAM)463,463,470                                                    05890
  463 IF(DT-TD(MM))464,464,470                                               05900
  464 LDFLAG=3                                                               05910
      DELAY=0.                                                               05920
      GO TO 331                                                              05930
  470 CALL TIMCON(VAL(I,1),TD(MM),TIP)                                       05940
      MM =  MM + 1                                                           05950
      CALL XTLEGN(XLCOD,L)                                                   05960
      PRINT 324,TIP,VAL(I,3),L                                               05970
  318 CONTINUE                                                               05980
      GO TO 9999                                                             05990
C  BEGIN DEVELOPMENT OF AIRCRAFT NARRATIVE                                   06000
  316 CALL XTLEGN(XLCOD,L)                                                   06010
      DO 34 K=1,NTOTLG                                                       06020
      NMINES = ACLDL(L,2)+1.                                                 06030
      DO267 NMIN = 1,NMINES                                                  06040
      DO 266  III=1,30                                                       06050
  266 OUTPUT(III)=0.                                                         06060
```

```
      IF(K-1)36,36,37                                                    06070
   36 IF(NMIN-1)38,38,37                                                 06080
   38 PRINT 39,VAL(I,1),VAL(I,3),L                                       06090
   39 FORMAT(F10.1,8H A/C NO.,F4.0,26H ARRIVES AT IP FOR LEG NO.,I4/)    06100
      GO TO 41                                                           06110
   37 IF(NMIN-NMINES)41,42,42                                            06120
   41 IF(DT-TD(MM))43,43,44                                              06130
   44 CALL TIMCON(VAL(I,1),TD(MM),TLAY)                                  06140
      NM = 3*(NMIN-1)+5                                                  06150
      NUBRS = VAL(I,5)                                                   06160
      NVEH = VAL(I,3)                                                    06170
C  SEARCH A/C BOMBRACKS FOR MINE THAT IS TO BE RELEASED                  06180
      DO 56 NUBR = 1,NUBRS                                               06190
      CALL XTRACT(BRTEMP(NVEH,NUBR),XTRACL,XTRACX,XTRACR)                06200
      IF(XTRACR-ACLDL(L,NM))56,57,56                                     06210
   56 CONTINUE                                                           06220
      PRINT 58,TLAY,ACLDL(L,NM)                                          06230
   58 FORMAT(F10.1,9H MINE NO.,F4.0,36H IS BLOCKED BY PRIOR RELEASE FAIL 06240
     1URE/)                                                              06250
      MLFLAG = 0                                                         06260
      GO TO 35                                                           06270
   57 CALL RAN1(IR,RN)                                                   06280
C  DETERMINES IF MINE RELEASES                                           06290
      IF(RN-ACCL(J,5))59,59,60                                           06300
C  IF MINE RELEASES,BOMBRACK ARRAY IS MODIFIED                           06310
   59 BRTEMP(NVEH,NUBR)=XTRACL                                           06320
C     PRINT 61,BRTEMP(NVEH,NUBR)                                         06330
C  61 FORMAT(//30H BOMBRACK LOAD AFTER RELEASE = ,F19.0)                 06340
      PRINT 265,TLAY,ACLDL(L,NM)                                         06350
  265 FORMAT(F10.1,9H MINE NO.,F4.0,7H LAYED /)                          06360
      MLFLAG =1                                                          06370
      GO TO 62                                                           06380
C  IF MINE DOES NOT RELEASE, BOMBRACK ARRAY IS NOT CHANGED               06390
   60 PRINT 63,TLAY,ACLDL(L,NM)                                          06400
   63 FORMAT(F10.1,9H MINE NO.,F4.0,16H DID NOT RELEASE/)                06410
      MLFLAG =0                                                          06420
```

```
      GO TO 35                                                          06430
   62 M1 = ACLDL(L,NM)                                                  06440
      MT2 = MDL(M1,2)                                                   06450
      CALL RAN1(IR,RN)                                                  06460
      IF(ACMCL(MT2,2)-RN)70,69,69                                       06470
   69 ECCENT = SQRTF((1.0-(ACMCL(MT2,3)-1.0)**2-1000.0**2)/(1.0-(ACMCL(M 06480
     1T2,3)-1.0)**2))                                                   06490
      CEP = (SQRTF(1.0-((ACCL(J,4))**2)/(1.0-ECCENT**2))+1.0)/3.0       06500
      SIGMA = CEP/1.1744                                                06510
C     PRINT 71,ECCENT,CEP,SIGMA                                         06520
C  71 FORMAT(//10H ECCENT = ,F20.10,5X,7H CEP = ,F20.10,5X,9H SIGMA = ,F 06530
C    120.10//)                                                          06540
      CALL RAN1(IR,RN)                                                  06550
      IF(ACMCL(MT2,5)-RN)72,72,73                                       06560
   73 OUTPUT(2)=1.                                                      06570
      GO TO 72                                                          06580
   70 OUTPUT(1)=1.                                                      06590
      ECCENT = SQRTF((1.0-(ACMCL(MT2,4)-1.0)**2-1000.0**2)/(1.0-(ACMCL(M 06600
     1T2,4)-1.0)**2))                                                   06610
      CEP = (SQRTF(1.0-((ACCL(J,4))**2)/(1.0-ECCENT**2))+1.0)/3.0       06620
      SIGMA = CEP/1.1744                                                06630
C     PRINT 71,ECCENT,CEP,SIGMA                                         06640
      CALL RAN1(IR,RN)                                                  06650
      IF(ACMCL(MT2,6)-RN)72,72,75                                       06660
   75 OUTPUT(2)=1.                                                      06670
C COMPUTATION OF XPOSIT AND YPOSIT                                      06680
   72 CALL RNORM(SIGMA,XE,IR)                                           06690
      CALL RNORM(SIGMA,YE,IR)                                           06700
      XPOSIT = ACLDL(L,NM+1) + XE                                       06710
      YPOSIT = ACLDL(L,NM+2) + YE                                       06720
      MFLAG = ACMCL(MT2,10)                                             06730
C     PRINT 74,XE,YE,XPOSIT,YPOSIT                                      06740
C  74 FORMAT(//6H XE = ,F10.4,6H YE = ,F10.4,10H XPOSIT = ,F10.4,10H YPO 06750
C    1SIT = ,F10.4//)                                                   06760
      CALL DEPTH (XPOSIT,YPOSIT,PHI,NMAX,MMAX,YMEANY,XMEANX,WD)         06770
      IF(WD)139,139,138                                                 06780
```

```
  139 OUTPUT(9) = 1.                                                        06790
      CALL TAPO(OUTPUT,TLAY,MT2,MDL(M1,3),MDL(M1,8),MDL(M1,7),MDL(M1,5),    06792
     1MDL(M1,6),M1,XPOSIT,YPOSIT,WD,MFLAG)                                  06794
      GO TO 35                                                              06800
  138 CALL TAPO(OUTPUT,TLAY,MT2,MDL(M1,3),MDL(M1,8),MDL(M1,7),MDL(M1,5),    06810
     1MDL(M1,6),M1,XPOSIT,YPOSIT,WD,MFLAG)                                  06820
C     PRINT 76,WD                                                         06830
C  76 FORMAT(/15H WATER DEPTH = ,E20.10,5H FEET /)                         06840
C  INSERT TIDE ROUTINE HERE IF USED                                       06850
      CALL DEPTH(XPOSIT,YPOSIT,BOT,NMAX1,MMAX1,YMEANY,XMEANX,BTMTYP)        06860
      GO TO(85,85,86,86,85,85),MFLAG                                        06870
   85 IF(WD-20.0)87,87,88                                                   06880
   87 IF(OUTPUT(1))89,89,90                                                 06890
   89 PDHB = ACMCL(MT2,7)                                                   06900
      GO TO 91                                                              06910
   90 PDHB = ACMCL(MT2,8)                                                   06920
   91 PD = (PDHB/5.)*BTMTYP                                                 06930
      CALL RAN1(IR,RN)                                                      06940
      IF(PD-RN)88,92,92                                                     06950
   92 OUTPUT(3)=1.                                                          06960
   88 CALL MOOR( OUTPUT,IR,MFLAG,WD,MDL(M1,4),ACMCL(MT2,12),ACMCL(MT2,13    06970
     1),ACMCL(MT2,14))                                                      06980
      GO TO 35                                                              06990
C  STATEMENT 86 BEGINS BOTTOM MINE LOOP.                                   07000
   86 PB = ACMCL(MT2,9)                                                     07010
      IF(WD-20.)194,194,195                                                 07020
  194 IF(OUTPUT(1))196,196,197                                              07030
  196 PB = PB + .1                                                          07040
      PDHB = ACMCL(MT2,7)                                                   07050
      GO TO 198                                                             07060
  197 PB = PB + .2                                                          07070
      PDHB = ACMCL(MT2,8)                                                   07080
  198 PD = (PDHB/ 5.)*BTMTYP                                                07090
      CALL RAN1(IR,RN)                                                      07100
      IF(PD-RN)200,200,195                                                  07110
  200 OUTPUT(3)=1.                                                          07120
```

```
  195 CALL BOTTOM (OUTPUT,IR,MFLAG,BTMTYP,PB,ACMCL(MT2,11),ACMCL(MT2,14)    07130
     1)                                                                    07140
   35 MM = MM + 1                                                          07150
      IF(MLFLAG)267,267,268                                                07160
  268 WRITE TAPE 3,OUTPUT                                                  07170
  267 CONTINUE                                                             07180
   43 CALL TIMCON(VAL(I,1),DT,TDT)                                         07190
      PRINT 46,TDT,VAL(I,3)                                                07200
   46 FORMAT(F10.1,8H A/C NO.,F3.0,40H ATTRITED. MINES REMAINING ON BOAR   07210
     1D ARE  )                                                            07220
   47 NVEH = VAL(I,3)                                                      07230
      NBR = VAL(I,5)                                                       07240
      DO 48 KK = 1,NBR                                                     07250
   50 CALL XTRACT(BRTEMP(NVEH,KK),XTRACL,XTRACX,XTRACR)                    07260
      PRINT 49,XTRACR                                                      07270
   49 FORMAT(58X,F4.0)                                                     07280
      BRTEMP(NVEH,KK)=XTRACL                                               07290
      IF(XTRACL)48,48,50                                                   07300
   48 CONTINUE                                                             07310
      GO TO 9999                                                           07320
C  DETERMINE IF A/C SURVIVES TRANSIT FROM TP TO IP FOR NEXT LEG            07330
   42 IF(DT-TD(MM))51,51,52                                                07340
   51 CALL TIMCON(VAL(I,1),DT,TDT)                                         07350
      PRINT 46,TDT,VAL(I,3)                                                07360
      GO TO 47                                                             07370
   52 CALL TIMCON(VAL(I,1),TD(MM),TTP)                                     07380
      MM = MM + 1                                                          07390
      PRINT 53,TTP,VAL(I,3),L                                              07400
   53 FORMAT(F10.1,8H A/C NO.,F4.0,26H ARRIVES AT TP FOR LEG NO.,I4//)     07410
      IF(K - NTOTLG) 54,9999,9999                                          07420
   54 IF(DT-TD(MM))51,55,55                                                07430
   55 CALL TIMCON (VAL(I,1),TD(MM),TIP)                                    07440
      MM = MM + 1                                                          07450
      CALL XTLEGN(XLCOD,L)                                                 07460
      PRINT 39,TIP,VAL(I,3),L                                              07470
   34 CONTINUE                                                             07480
```

```
9999 CONTINUE                                                          07490
8500 CONTINUE                                                          07500
     END FILE 3                                                        07510
     END                                                               07520
```

```
      SUBROUTINE XTLEGN(XLCOD,LEGNUM)                                    07530
      DIMENSION XLCOD(6)                                                 07540
      IF(XLCOD(1)) 1,1,3                                                 07550
    1 DO 2 I=1,5                                                         07560
    2 XLCOD(I) = XLCOD(I + 1)                                            07570
      XLCOD(6) = 000000000.0                                             07580
    3 LEGNUM = XLCOD(1) / 1000000.0                                      07590
      XLGNUM = LEGNUM                                                    07600
      XLCOD(1) = (XLCOD(1) - XLGNUM * 1000000.0) * 1000.0                07610
      RETURN                                                             07620
      END                                                                07630
```

```
      SUBROUTINE XTRACT(XNUMBR,XTRACL,XTRACX,XTRACR)                    07640
      NXTRCL = XNUMBR / 1000.0                                          07650
      XTRACL = NXTRCL                                                   07660
      XTRACX = XTRACL * 1000.                                           07670
      XTRACR = XNUMBR - XTRACX                                          07680
      RETURN                                                            07690
      END                                                               07700
```

```
      SUBROUTINE TIMCON (ARTDHM,TD,TF)                                  07710
      ND=ARTDHM/ 10000.0                                                07720
      AND = ND                                                          07730
      ARTHM = ARTDHM - AND * 10000.0                                    07740
      NH = ARTHM/100.0                                                  07750
      ANH = NH                                                          07760
      ARTM = ARTHM - ANH * 100.0                                        07770
       TEMT = ARTM + TD                                                 07780
      NTEH = TEMT / 60.0                                                07790
      ANTEH = NTEH                                                      07800
      TEM = TEMT - ANTEH * 60.0                                         07810
      NTED =(ANTEH +ANH)/24.0                                           07820
      ANTED=NTED                                                        07830
      TEH=ANTEH + ANH - ANTED*24.0                                      07840
      TF = TEM + TEH * 100.0 + (ANTED + AND) * 10000.0                  07850
      RETURN                                                            07860
      END                                                               07870
```

```
      SUBROUTINE RNORM(SIGMA,RANDOM,IR)                                 07880
      DIMENSION RAN(3)                                                  07890
      SUMRAN=0.                                                         07900
      DO 1 I=1,3                                                        07910
      CALL RAN1 (IR,RAN(I))                                             07920
    1 SUMRAN=SUMRAN+RAN(I)                                              07930
      RANDOM=2.0*SIGMA*(SUMRAN-1.5)                                     07940
      RETURN                                                            07950
      END                                                               07960
```

```
      SUBROUTINE RAN1(IY,ANS)                                          07970
      A=0.                                                             07980
      B=1.                                                             07990
      IY = 3125 * IY                                                   08000
      IY = IY- (IY/67108864) * 67108864                                08010
      FY = IY                                                          08020
      ANS  = FY/67108864. *(B-A) + A                                   08030
      RETURN                                                           08040
      END                                                              08050
```

```
      SUBROUTINE DEPTH(A,B,POLYCO,I,J,C,D,ZCOMP)                        08060
      DIMENSION POLYCO(45,45)                                           08070
      ZCOMP = 0.                                                        08080
      IS = I                                                            08090
  548 POLY = POLYCO(IS,J)                                               08100
      IT = J-1                                                          08110
  549 POLY = POLY*(B-C)+POLYCO(IS,IT)                                   08120
      IF(IT-1)547,547,542                                               08130
  542 IT = IT-1                                                         08140
      GO TO 549                                                         08150
  547 ZCOMP = ZCOMP*(A-D)+POLY                                          08160
      IF(IS-1)550,550,544                                               08170
  544 IS = IS-1                                                         08180
      GO TO 548                                                         08190
  550 RETURN                                                            08200
      END                                                               08210
```

```
      SUBROUTINE MOOR(XOUT,IIR,MXFLAG,XWD,MDLX,CASEP,FULDEP,REL)          08220
      DIMENSION XOUT(30)                                                  08230
      CALL RAN1(IIR,RN)                                                   08240
      IF(CASEP-RN)93,94,94                                                08250
   93 XOUT(4) = 1.                                                        08260
      GO TO 95                                                            08270
   94 CALL RAN1(IIR,RN)                                                   08280
      IF(FULDEP-RN)96,97,97                                               08290
   96 XOUT(5) = 1.                                                        08300
      CALL RAN1(IIR,DPLP)                                                 08310
      XMDL=MDLX                                                           08320
      GO TO 98                                                            08330
   97 DPLP = 1.0                                                          08340
      XMDL=MDLX                                                           08350
   98 GO TO(115,116,117,117,115,116),MXFLAG                               08360
  117 PRINT 118,MXFLAG                                                    08370
  118 FORMAT(22H ERROR STOP. MXFLAG = ,I3//)                              08380
      GO TO 35                                                            08390
  115 ACTDPL = (XWD - XMDL)*DPLP                                          08400
C     PRINT 2005,ACTDPL                                                   08410
C2005 FORMAT(/10H ACTDPL = ,F10.3/)                                       08420
      GO TO 119                                                           08430
  116 ACTDPL = XMDL * DPLP                                                08440
  119 CD = XWD - ACTDPL                                                   08450
      XOUT(20) = CD                                                       08460
      IF(CD)135,135,95                                                    08470
  135 XOUT(11) = 1.                                                       08480
   95 GO TO(120,120,127,127,128,128),MXFLAG                               08490
  127 PRINT 118,MXFLAG                                                    08500
      GO TO 35                                                            08510
  128 IF(XOUT(4)) 120,120,178                                             08520
  178 XOUT(10) = 1.                                                       08530
      GO TO 35                                                            08540
  120 CALL RAN1(IIR,RN)                                                   08550
      IF(REL- RN)129,35,35                                                08560
  129 XOUT(8)=1.                                                          08570
   35 RETURN                                                              08580
      END                                                                 08590
```

```
      SUBROUTINE BOTTOM(XOUT,IIR,MXFLAG,XBTM,XPB,XNWL,REL)                08600
      DIMENSION XOUT(30)                                                  08610
      XPB = XPB - (XPB/5.)* XBTM                                          08620
      GO TO(201,201,202,203,201,201),MXFLAG                               08630
  201 PRINT 204,MXFLAG                                                    08640
  204 FORMAT(22H ERROR STOP. MXFLAG = ,I3//)                              08650
      GO TO 35                                                            08660
  202 CALL RAN1(IIR,RN)                                                   08670
      IF(XPB - RN) 205,205,206                                            08680
  206 XOUT(22) = XNWL                                                     08690
      XOUT(6)=1.                                                          08700
      GO TO 205                                                           08710
  203 CALL RAN1(IIR,RN)                                                   08720
      IF(XPB - RN) 207,207,205                                            08730
  207 XOUT(22)=XNWL                                                       08740
      XOUT(7)=1.                                                          08750
  205 CALL RAN1(IIR,RN)                                                   08760
      IF(REL -RN)226,35,35                                                08770
  226 XOUT(8)=1.                                                          08780
   35 RETURN                                                              08790
      END                                                                 08800
```

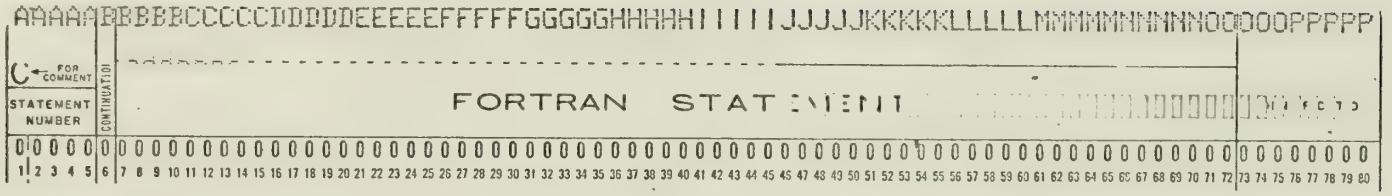
```
      SUBROUTINE TIMDIF(DHM1,DHM2,TDIF)                                 08810
      DIMENSION DHM(2)                                                  08820
      DHM(1)=DHM1                                                       08830
      DHM(2)=DHM2                                                       08840
      DO 1 I=1,2                                                        08850
      ND=DHM(I)/10000.                                                  08860
      AND=ND                                                            08870
      NH=(DHM(I)-AND*10000.)/100.                                       08880
      ANH=NH                                                            08890
    1 DHM(I)=DHM(I)-AND*10000.-ANH*100.+ANH*60.+AND*1440.               08900
      TDIF=DHM(2)-DHM(1)                                                08910
      RETURN                                                            08920
      END                                                               08930
```

```
      SUBROUTINE POSIT(IR,B,A,G,S1,S2,D,E,X,Y)                    08940
      CALL RAN1(IR,RN)                                            08950
      THETA = 6.28318*RN                                          08960
      XMU = (A-B)/(G-B)                                           08970
      CALL RAN1(IR,RN)                                            08980
      IF(RN-XMU) 1,1,2                                            08990
    2 RERROR = SQRTF((G-A)*(G-B)*(1.-RN))                         09000
      GO TO 3                                                     09010
    1 RERROR = B + SQRTF(RN*(G-B)*(A-B))                          09020
    3 X=S1 + RERROR*COSF(THETA) + D                               09030
      Y = S2 + RERROR*SINF(THETA) +E                              09040
      RETURN                                                      09050
      END                                                         09060
```

```
      SUBROUTINE TAPO(O,TIME,TYP,M1,M2,M3,M4,M5,NO,X,Y,W,MF)          09070
      DIMENSION O(30)                                                09080
      O(16)=TIME                                                     09090
      O(17)=NO                                                       09100
      O(18)=X                                                        09110
      O(19)=Y                                                        09120
      O(20)=W                                                        09130
      O(21) = TYP                                                    09140
      O(22) = M1                                                     09150
      O(23)=M2                                                       09160
      O(24)=M3                                                       09170
      O(25)= M4                                                      09180
      O(26)=M5                                                       09190
      O(27)=MF                                                       09200
      RETURN                                                         09210
      END                                                            09220
```

INPUT FORMAT

CARD NO. ONE
NUMBER OF CARDS REQUIRED = ONE

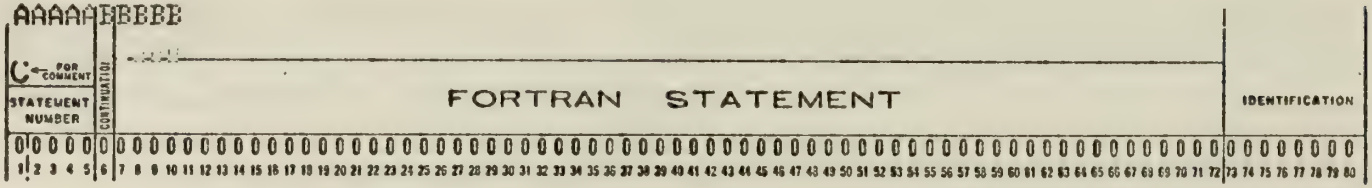



| | | | |
|---|---|---|---|
| A. | I5 | NV | TOTAL NUMBER OF VEHICLES. |
| B. | I5 | NA | TOTAL NUMBER OF AIRCRAFT |
| C. | I5 | NAT | TOTAL NUMBER OF AIRCRAFT TYPES. |
| D. | I5 | NACR | NUMBER OF ROWS IN AIRCRAFT LEG DESCRIPTION LIST.(SAME AS TOTAL NUMBER OF LEGS ASSIGNED AIRCRAFT) |
| E. | I5 | NACC | NUMBER OF COLUMNS IN AIRCRAFT LEG DESCRIPTION LIST.(SAME AS THE LEG WITH GREATEST NUMBER OF MINES ON IT AND EQUALS THE NUMBER OF MINES ON THAT LEG TIMES THREE PLUS SEVEN) |
| F. | I5 | MACT | NUMBER OF AIRCRAFT DELIVERED MINE TYPES. |
| G. | I5 | NSH | TOTAL NUMBER OF SHIPS. |
| H. | I5 | NSHT | TOTAL NUMBER OF SHIP TYPES. |
| I. | I5 | NSHR | NUMBER OF ROWS IN SHIP LEG DESCRIPTION LIST.(SAME AS TOTAL NUMBER OF LEGS ASSIGNED SHIPS) |
| J. | I5 | NSHC | NUMBER OF COLUMNS IN SHIP LEG DESCRIPTION LIST.(SAME AS THE LEG WITH THE GREATEST NUMBER OF MINES ON IT AND EQUALS THE NUMBER OF MINES ON THAT LEG TIMES THREE PLUS SEVEN) |

| | | | |
|---|---|---|---|
| K. | I5 | MSHT | NUMBER OF SHIP DELIVERED MINE TYPES. |
| L. | I5 | NSU | TOTAL NUMBER OF SUBMARINES. |
| M. | I5 | NSUT | TOTAL NUMBER OF SUBMARINE TYPES. |
| N. | I5 | NSUR | NUMBER OF ROWS IN SUBMARINE LEG DESCRIPTION LIST.(SAME AS TOTAL NUMBER OF LEGS ASSIGNED SUBMARINES) |
| O. | I5 | NSUC | NUMBER OF COLUMNS IN SUBMARINE LEG DESCRIPTION LIST.(SAME AS THE LEG WITH GREATEST NUMBER OF MINES ON IT AND EQUALS THE NUMBER OF MINES ON THAT LEG TIMES FIVE PLUS SEVEN) |
| P. | I5 | MSUT | NUMBER OF SUBMARINE DELIVERED MINE TYPES. |

NOTE-IF ANY OF THE FOREGOING INDICES EXCEEDS ITS CORRESPONDING VALUE IN THE PROGRAM DIMENSION STATEMENT,A PROGRAM CHANGE TO THE DIMENSION AND COMMON STATEMENTS IS REQUIRED.FURTHER IF THE TOTAL NUMBER OF MINES TO BE LAYED BY ANY SINGLE VEHICLE PLUS TWO TIMES THE NUMBER OF LEGS ASSIGNED THAT VEHICLE EXCEEDS THE DIMENSION OF THE TD ARRAY,THEN THE TD ARRAY MUST BE INCREASED TO EXCEED THAT NUMBER,AND THE UPPER INDICES ON THE DO LOOPS AT ONE STATEMENT BEFORE STATEMENT NO. 346 AND AT STATEMENT NO. 594 MUST BE ASSIGNED THAT DIMENSION VALUE MINUS ONE.

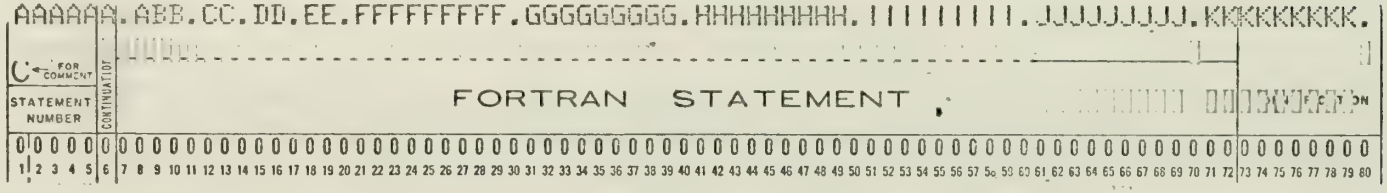
CARD NO. TWO
NUMBER OF CARDS REQUIRED = ONE




| | | | |
|---|---|---|---|
| A. | I5 | MINTOT | TOTAL NUMBER OF MINES. |
| B. | I6 | NF | NUMBER OF POINTS USED TO DEFINE MINEFIELD PERIMETER. |

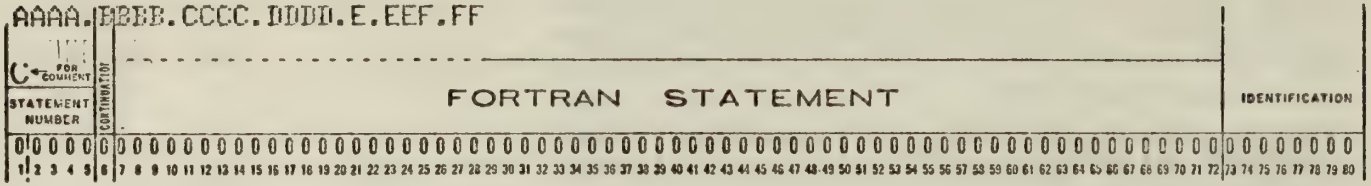
NOTE-IF ANY OF THE FOREGOING INDICES EXCEEDS ITS CORRESPONDING VALUE IN THE PROGRAM DIMENSION STATEMENT, A PROGRAM CHANGE TO THE DIMENSION AND COMMON STATEMENTS IS REQUIRED.

CARD NO. THREE
NUMBER OF CARDS REQUIRED = TOTAL NUMBER OF VEHICLES (NV) IN CHRONOLOGICAL ORDER.
ARRAY NAME = VEHICLE ARRIVAL LIST (VAL,NV BY 11)




| | | | |
|---|---|---|---|
| A. | F8.1 | VAL(I,1) | DATE-TIME OF ARRIVAL OF ITH VEHICLE TO NEAREST TENTH OF A MINUTE.(EX.,080230.1) |
| B. | F3.0 | VAL(I,2) | VEHICLE TYPE INDICATOR.(EX.,000-009 ARE SUBMARINE TYPES,010-099 ARE SHIP TYPES, 100-999 ARE AIRCRAFT TYPES) |
| C. | F3.0 | VAL(I,3) | VEHICLE NUMBER WITHIN A GENERAL VEHICLE TYPE.(EX.,AIRCRAFT NO.1,SHIP NO.3,SUBMARINE NO.1) |
| D. | F3.0 | VAL(I,4) | TOTAL NUMBER OF LEGS ASSIGNED THIS VEHICLE.(EX.,02 IMPLIES TWO LEGS ASSIGNED) |
| E. | F3.0 | VAL(I,5) | NUMBER OF LOADED BOMBRACKS ON THIS AIRCRAFT. (EX., 04 IMPLIES FOUR LOADED BOMBRACKS. 000 IS INSERTED FOR SHIPS AND SUBMARINES) |
| F. THRU K. | F10.0 EACH | VAL(I,6) THRU VAL(I,11) | CODED LEG ASSIGNMENT LIST,STARTING WITH LEFT ELEMENT LEG NUMBERS ARE ASSIGNED IN THE ORDER THEY ARE TO BE TRANSVERSED BY THREE DIGIT SUB-GROUPS READING FROM LEFT TO RIGHT(EX.,VAL(I,6)= 002003006. AND VAL(I,7)= 25001000. IMPLIES THAT THE VEHICLE WILL TRANSIT LEGS NO. 2, 3, 6, 25, AND 1 IN THAT ORDER) |

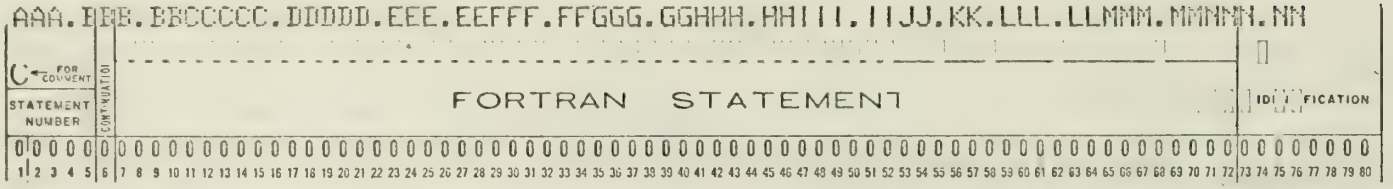
CARD NO. FOUR
NUMBER OF CARDS REQUIRED = TOTAL NUMBER OF AIRCRAFT TYPES (NAT) IN CONSECUTIVE ORDER OF AIRCRAFT TYPE NUMBER
ARRAY NAME = AIRCRAFT CHARACTERISTICS LIST (ACCL,NAT BY 6)




| | | | |
|---|---|---|---|
| A. | F5.0 | ACCL(J,1) | AIRCRAFT TYPE NUMBER. A THREE DIGIT NUMBER FROM 100 TO 999 INCLUSIVE (EX.,101). |
| B. | F5.0 | ACCL(J,2) | AIRCRAFT TRANSIT SPEED IN KNOTS FOR THIS AIRCRAFT TYPE. |
| C. | F5.0 | ACCL(J,3) | AIRCRAFT MINELAYING SPEED IN KNOTS FOR THIS AIRCRAFT TYPE. |
| D. | F5.0 | ACCL(J,4) | AIRCRAFT MINELAYING ALTITUDE IN FEET FOR THIS AIRCRAFT TYPE. |
| E. | F4.2 | ACCL(J,5) | PROBABILITY OF MINE RELEASE FOR THIS AIRCRAFT TYPE. |
| F. | F4.2 | ACCL(J,6) | PROBABILITY OF SURVIVAL OVER OBJECTIVE AREA FOR THIS AIRCRAFT TYPE. |

CARD NO. FIVE
NUMBER OR CARDS REQUIRED = TOTAL NUMBER OF AIRCRAFT MINE TYPES (MACT) IN CONSECUTIVE ORDER OF MINE TYPE NUMBER

ARRAY NAME = AIRCRAFT MINE CHARACTERISTICS LIST (ACMCL, MACT BY 14)

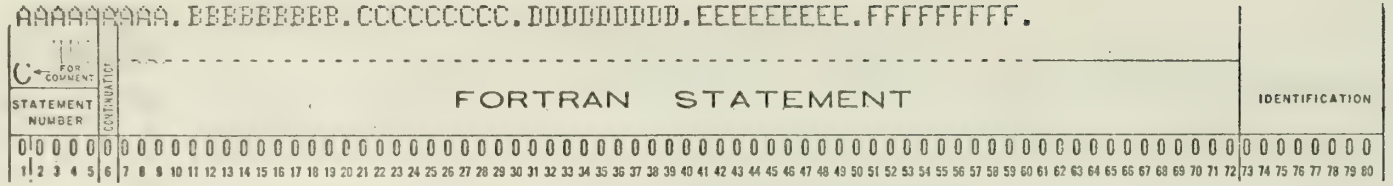



| | | | |
|---|---|---|---|
| A. | F4.0 | ACMCL(J,1) | MINE TYPE NUMBER. |
| B. | F6.2 | ACMCL(J,2) | PROBABILITY OF CHUTE DEPLOYMENT FOR PARACHUTE EQUIPPED MINES. FOR THIS MINE TYPE. FOR NON-PARACHUTE EQUIPPED MINES INSERT 1.0. |
| C. | F6.0 | ACMCL(J,3) | CEP IN YARDS FOR AN ALTITUDE OF 3000 FEET GIVEN PARACHUTE DEPLOYS FOR THIS MINE TYPE THIS IS ALSO CEP FOR NON-CHUTE EQUIPPED MINES. |
| D. | F6.0 | ACMCL(J,4) | CEP IN YARDS FOR AN ALTITUDE OF 3000 FEET GIVEN PARACHUTE DOES NOT DEPLOY FOR THIS MINE TYPE. INSERT 00. FOR NON-CHUTE EQUIPPED MINES. |
| E. | F6.2 | ACMCL(J,5) | PROBABILITY OF WATER IMPACT DAMAGE GIVEN CHUTE DEPLOYS FOR THIS MINE TYPE. THIS IS ALSO PROBABILITY OF WATER IMPACT DAMAGE FOR NON-CHUTE EQUIPPED MINES. |
| F. | F6.2 | ACMCL(J,6) | PROBABILITY OF WATER IMPACT DAMAGE GIVEN CHUTE DOES NOT DEPLOY FOR THIS MINE TYPE. INSERT 0.0 FOR NON-CHUTE EQUIPPED MINES. |
| G. | F6.2 | ACMCL(J,7) | PROBABILITY OF IMPACT DAMAGE ON THE HARDEST BOTTOM GIVEN THE WATER DEPTH IS LESS THAN 20 FEET AND CHUTE DEPLOYS FOR THIS MINE TYPE. THIS IS ALSO THE PROBABILITY OF IMPACT DAMAGE ON THE HARDEST BOTTOM GIVEN THE WATER DEPTH IS LESS THAN 20 FEET FOR NON-CHUTE EQUIPPED MINES. |
| H. | F6.2 | ACMCL(J,8) | PROBABILITY OF IMPACT DAMAGE ON THE HARDEST BOTTOM GIVEN THE WATER DEPTH IS LESS THAN 20 FEET AND CHUTE DOES NOT DEPLOY FOR THIS MINE TYPE. INSERT 0.00 FOR NON-CHUTE EQUIPPED MINES. |

| | | | |
|---|---|---|---|
| I. | F6.2 | ACMCL(J,9) | PROBABILITY OF BURYING IN THE SOFTEST BOTTOM FOR THIS MINE TYPE. |
| J. | F3.0 | ACMCL(J,10) | INDICATOR OF BOTTOM OR MOOR TYPE HAVING SIX POSSIBLE VALUES WHERE 1 AND 5 ARE MOORED OF FIXED CABLE LENGTH TYPE, 2 AND 6 ARE MOORED OF FIXED CASE DEPTH TYPE BUT 5 AND 6 DO NOT ARM UNLESS CASE AND ANCHOR SEPARATION TAKES PLACE, 3 IS A BOTTOM MINE FOR WHICH BURYING IS UNDESIRABLE, AND 4 IS A BOTTOM MINE WHERE BURYING IS DESIREABLE. |
| K. | F3.0 | ACMCL(J,11) | MINE TYPE I (NWL TYPE), KEY TO ACTUATION CURVES FOR MINES BURYING OR NOT BURYING WHICHEVER IS DESIREABLE FOR BOTTOM MINES OF THIS TYPE AND SHOULD REFLECT A DIMINISHED CAPABILITY. INSERT ZERO FOR MOORED MINES. |
| L. | F6.2 | ACMCL(J,12) | PROBABILITY OF CASE AND ANCHOR SEPARATION FOR MOORED MINES ONLY. |
| M. | F6.2 | ACMCL(J,13) | PROBABILITY OF FULL ANCHOR CABLE DEPLOYMENT FOR MOORED MINES ONLY. |
| N. | F6.2 | ACMCL(J,14) | PROBABILITY THAT THE MINE WILL OPERATE GIVEN IT HAS NOT BEEN DAMAGED, I.E., THE MINES RELIABILITY. |

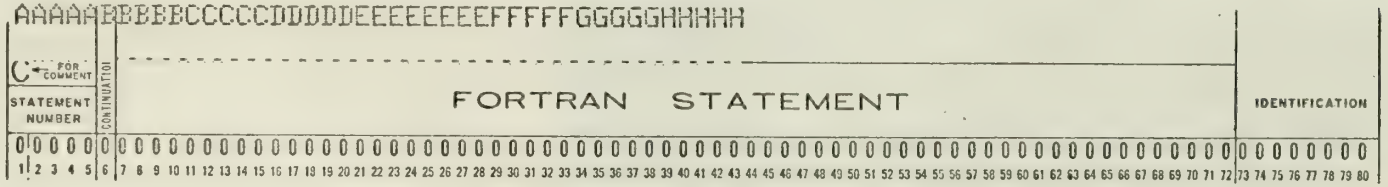
CARD NO. SIX
NUMBER OF CARDS REQUIRED = TOTAL NUMBER OF AIRCRAFT (NA) IN CONSECUTIVE ORDER OF AIRCRAFT NUMBER.

ARRAY NAME = BOMBRACK ARRAY (BR,NA BY 6)




| | | | |
|---|---|---|---|
| A. THRU F. | F10.0 EACH | BR(I,1) THRU BR(I,6) | ORDERED LIST OF THE MINES IN EACH OF SIX POSSIBLE AIRCRAFT BOMBRACKS BY THREE-DIGIT GROUPS FOR EACH MINE NUMBER IN ORDER OF RELEASE FROM RIGHT TO LEFT (EX.,001002027. IMPLIES MINES ARE STORED IN BOMBRACK IN TH ORDER 27,2,1 AND CAN ONLY BE RELEASED IN THAT ORDER. |

CARD NO. SEVEN
NUMBER OF CARDS REQUIRED = TOTAL NUMBER OF MINES (MINTOT) IN CONSECUTIVE ORDER OF MINE NUMBER.
ARRAY NAME = MINE DESCRIPTION LIST (MDL, MINTOT BY 8)

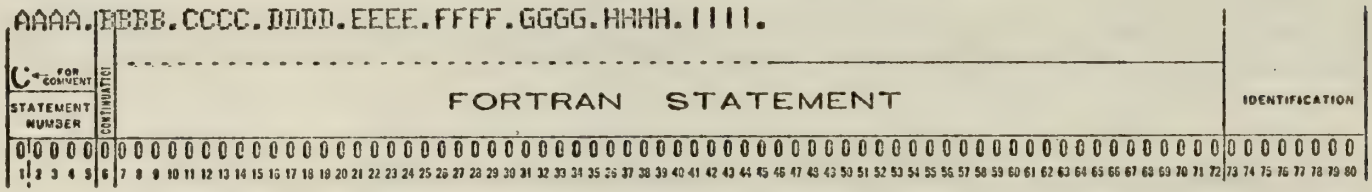



| | | | |
|---|---|---|---|
| A. | I5 | MDL(J,1) | MINE CASE NUMBER. |
| B. | I5 | MDL(J,2) | MINE TYPE II CORRESPONDS TO J INDEX TO RESPECTIVE MINE CHARACTERISTIC LIST. (EX., FOR A MINE WHICH IS AIRCRAFT LAYED AND IS TYPE NO. TWO ON THE AIRCRAFT MINE CHARAC-TERISTICS LIST, MDL(J,2) = 2.) |

| | | | |
|---|---|---|---|
| C. | I5 | MDL(J,3) | MINE TYPE I, SPACE SAVED FOR INSERTING A KEY TO ACTUATION CURVES. |
| D. | I5 | MDL(J,4) | CASE DEPTH, THE SETTING OF FEET BELOW SURFACE FOR FIXED CASE DEPTH MOORED MINES OR CABLE LENGTH (HEIGHT ABOVE BOTTOM) FOR FIXED CABLE LENGTH MOORED MINES. THIS VALUE IS ZERO FOR BOTTOM MINES. |
| E. | I8 | MDL(J,5) | ARMING DELAY IN MINUTES FOR THIS MINE. |
| F. | I5 | MDL(J,6) | STERILE TIME IN DAYS FOR THIS MINE. |
| G. | I5 | MDL(J,7) | SHIP COUNT SETTING FOR THIS MINE. |
| H. | I5 | MDL(J,8) | MOORING CABLE TYPE, KEY TO TYPE OF MOORING CABLE FOR MOORED MINES, I.E., 1 INDICATES ONE-HALF INCH WIRE CABLE AND 5 MIGHT INDICATE CHAIN FOR THIS MINE. |

NOTE - EXCEPT FOR ITEMS A, B, C, D, AND E THE ITEMS ABOVE ARE NOT UTILIZED IN MINDEL AND THEY ARE INCLUDED ONLY AS A BOOKKEEPING FEATURE FOR THEIR USE IN ANY SUBSEQUENT THREAT ASSESSMENT MODEL.

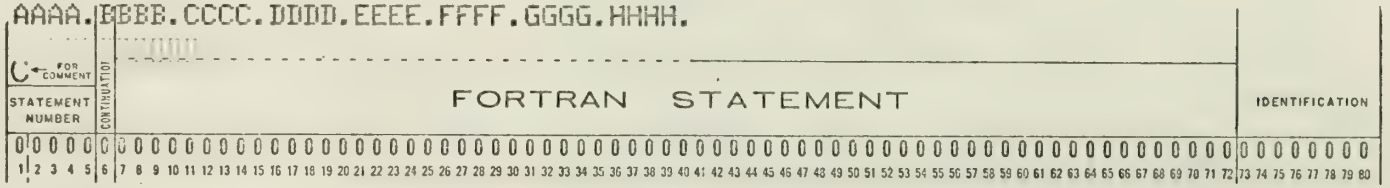
CARD NO. EIGHT
NUMBER OF CARDS REQUIRED = TOTAL NUMBER OF SHIP TYPES (NSHT) IN CONSECUTIVE ORDER OF SHIP TYPE NUMBER.
ARRAY NAME = SHIP CHARACTERISTIC LIST (SHCL,NSHT BY 9)




| | | | |
|---|---|---|---|
| A. | F5.0 | SHCL(J,1) | SHIP TYPE NUMBER |
| B. | F5.0 | SHCL(J,2) | TRANSIT SPEED IN KNOTS FOR THIS SHIP TYPE. |
| C. | F5.0 | SHCL(J,3) | MINELAYING SPEED IN KNOTS FOR THIS SHIP TYPE. |
| D. | F5.0 | SHCL(J,4) | ESTIMATE OF LEAST NAVIGATIONAL ERROR IN YARDS FOR THIS SHIP TYPE. |

| | | | |
|---|---|---|---|
| E. | F5.0 | SHCL(J,5) | ESTIMATE OF AVERAGE NAVIGATIONAL ERROR IN YARDS FOR THIS SHIP TYPE. |
| F. | F5.0 | SHCL(J,6) | ESTIMATE OF GREATEST NAVIGATIONAL ERROR IN YARDS FOR THIS SHIP TYPE. |
| G. | F5.0 | SHCL(J,7) | PROBABILITY OF RELEASING A MINE ON THE FIRST ATTEMPT FOR THIS SHIP TYPE. |
| H. | F5.0 | SHCL(J,8) | MAXIMUM TIME DELAY UNTIL MINE RELEASE AFTER A FAILURE ON FIRST ATTEMPT FOR THIS SHIP TYPE. |
| I. | F5.0 | SHCL(J,9) | PROBABILITY OF SUFFERING SOME DAMAGE FOR THE FIRST HALF-HOUR IN THE MINELAYING OBJECTIVE AREA FOR THIS SHIP TYPE. |

CARD NO. NINE
NUMBER OF CARDS REQUIRED = TOTAL NUMBER OF SHIP MINE TYPES (MSHT) IN CONSECUTIVE ORDER OF MINE TYPE NUMBER.
ARRAY NAME = SHIP MINE CHARACTERISTIC LIST(SHMCL, MSHT BY 8)

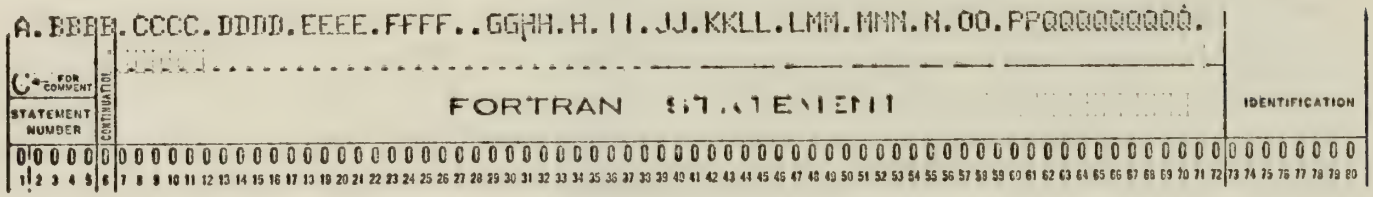



| | | | |
|---|---|---|---|
| A. | F5.0 | SHMCL(J,1) | MINE TYPE NUMBER |
| B. | F5.0 | SHMCL(J,2) | PROBABILITY OF IMPACT DAMAGE ON THE HARDEST BOTTOM GIVEN THE WATER DEPTH LESS THAN 20 FEET FOR THIS MINE TYPE. |
| C. | F5.0 | SHMCL(J,3) | PROBABILITY OF BURYING IN THE SOFTEST BOTTOM FOR BOTTOM MINES OF THIS TYPE. INSERT ZERO FOR MOORED MINES. |
| D. | F5.0 | SHMCL(J,4) | INDICATOR OF BOTTOM OR MOOR TYPE MINE. SEE ITEM J OF ACMCL. |

| | | |
|---|---|---|
| E. | F5.0 | SHMCL(J,5) MINE TYPE I(NWL TYPE), KEY TO ACTUATION CURVES FOR MINES BURYING OR NOT BURYING WHICHEVER IS UNDESIRABLE FOR BOTTOM MINES OF THIS TYPE AND SHOULD REFLECT A DIMINISHED CAPABILITY. INSERT ZERO FOR MOORED MINES. |
| F. | F5.0 | SHMCL(J,6) PROBABILITY OF CASE AND ANCHOR SEPARATION FOR MOORED MINES OF THIS TYPE. INSERT ZERO FOR BOTTOM MINES. |
| G. | F5.0 | SHMCL(J,7) PROBABILITY OF FULL ANCHOR CABLE DEPLOYMENT FOR MOORED MINES OF THIS TYPE. INSERT ZERO FOR BOTTOM MINES. |
| H. | F5.0 | SHMCL(J,8) PROBABILITY THAT THE MINE WILL OPERATE GIVEN IT HAS NOT BEEN DAMAGED, I.E., THE MINES RELIABILITY. |



CARD NO. TEN
NUMBER OF CARDS REQUIRED = TOTAL NUMBER OF SUBMARINE TYPES (NSUT) IN CONSECUTIVE ORDER OF SUBMARINE TYPE NUMBER.
ARRAY NAME = SUBMARINE CHARACTERISTICS LIST(SUCL, NSUT BY 17)

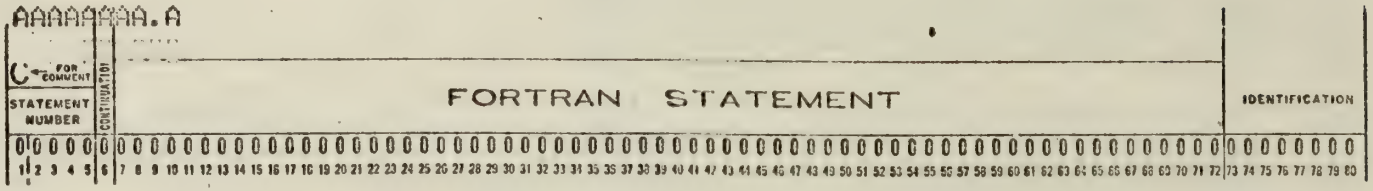



| | | |
|---|---|---|
| A. | F2.0 | SUCL(J,1) SUBMARINE TYPE NUMBER. |
| B. | F5.0 | SUCL(J,2) SUBMARINE TRANSIT SPEED IN KNOTS FOR THIS SUBMARINE TYPE. |
| C. | F5.0 | SUCL(J,3) SUBMARINE MINELAYING SPEED IN KNOTS FOR THIS SUBMARINE TYPE. |
| D. | F5.0 | SUCL(J,4) ESTIMATE OF LEAST NAVIGATIONAL ERROR FOR THIS SUBMARINE TYPE. |
| E. | F5.0 | SUCL(J,5) ESTIMATE OF AVERAGE NAVIGATIONAL ERROR FOR THIS SUBMARINE TYPE. |

| | | | |
|---|---|---|---|
| F. | F5.0 | SUCL(J,6) | ESTIMAT OF GREATEST NAVIGATIONAL ERROR FOR THIS SUBMARINE TYPE. |
| G. | F3.2 | SUCL(J,7) | PROBABILITY OF THE MINE BEING READY FOR FIRING FOR THIS SUBMARINE TYPE. |
| H. | F4.1 | SUCL(J,8) | MAXIMUM TIME DELAY TO BE ALLOWED BEFORE SUBMARINE SKIPS THE LAYING OF A MINE FOR THIS SUBMARINE TYPE. |
| I. | F3.2 | SUCL(J,9) | PROBABILITY OF DETECTION IN THE MINELAYING OBJECTIVE AREA FOR THIS SUBMARINE TYPE. |
| J. | F3.2 | SUCL(J,10) | PROBABILITY OF KNOWLEDGE OF DETECTION GIVEN A DETECTION FOR THIS SUBMARINE TYPE. SHOULD REFLECT THE CHARACTERISTICS OF DETECTION SYSTEMS PRESENT. |
| K. | F3.2 | SUCL(J,11) | PROBABILITY OF KNOWLEDGE OF DETECTION GIVEN NO DETECTION FOR THIS SUBMARINE TYPE. SHOULD REFLECT THE LIKLIHOOD THAT THE SUBMARINE WILL CONSIDER ITSELF DETECTED WHEN IN FACT IT WAS NOT DETECTED. |
| L. | F4.1 | SUCL(J,12) | PROBABILITY OF SUSPENDING OPERATIONS GIVEN KNOWLEDGE OF DETECTION FOR THIS SUBMARINE TYPE. REFLECTS THE OPERATIONAL DOCTRINE DETECTED (= 1.0) OR TO CONTINUE OPS AT ALL COSTS (= 0.0). |
| M. | F4.1 | SUCL(J,13) | PROBABILITY OF RETURNING TO MISSION GIVEN THAT THE MISSION HAS BEEN SUSPENDED FOR THIS SUBMARINE TYPE. REFLECTS THE OPERATIONAL DOCTRINE REQUIRING THE SUBMARINE TO RETURN AFTER SUSPENDING OPS (= 1.0) OR TO TERMINATE THE MISSION (= 0.0). |
| N. | F4.1 | SUCL(J,14) | PROBABILITY OF RETURNING TO MISSION GIVEN THAT THE SUBMARINES SURVIVES AN ATTACK FOR THIS SUBMARINE TYPE. REFLECTS THE OPERATIONAL DOCTRINE REQUIRING THE SUBMARINE TO RETURN AFTER SURVIVING AN ATTACK ( = 1.0) OR TERMINATE THE MISSION (= 0.0). |

| | | | |
|---|---|---|---|
| O. | F3.2 | SUCL(J,15) | PROBABILITY OF AN ATTACK GIVEN A DETECTION REFLECTS THE LIKLIHOOD OF AN ATTACK FOLLOWING A DETECTION. |
| P. | F3.2 | SUCL(J,16) | PROBABILITY OF A SUCCESSFUL ATTACK GIVEN AN ATTACK. |
| Q. | F10.0 | SUCL(J,17) | APPROXIMATE DISTANCE FROM MINEFIELD CENTER TO ONE HUNDRED FATHOM CURVE. |

CARD NO. ELEVEN
NUMBER OF CARDS REQUIRED = ONE




| | | | |
|---|---|---|---|
| A. | F10.1 | TAM. | DATE-TIME IN DAYS-HOURS-MINUTES TO NEAREST TENTH BY WHICH SUBMARINE OPERATIONS ARE TO BE TERMINATED. IF ONLY SUBMARINES ARE IN PLAY OR SUBMARINES ARE NOT FIRST IN THE OPERATION AREA, THEN A VALUE MUST BE PROVIDED. OTHERWISE A ZERO VALUE MAY BE INSERTED AND TAM IS SET BY THE PROGRAM TO THE TIME OF ARRIVAL OF THE NEXT VEHICLE WHICH IS NOT A SUBMARINE. |

CARD NO. TWELVE
NUMBER OF CARDS REQUIRED = TOTAL NUMBER OF SUBMARINE MINE TYPES (MSUT) IN CONSECUTIVE ORDER OF MINE TYPE NUMBER.

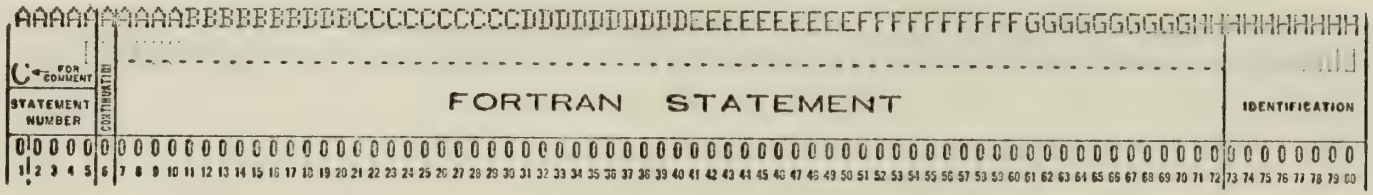
ARRAY NAME = SUBMARINE MINE CHARACTERISTICS LIST (SUMCL, MSUT BY 13)

```
AAAA.BB.BBCC.CCDD.EEEE.FF.FFGG.GGHH.HHII.IIJJ.JJKK.KKLLLL.MM.MM
C← FOR COMMENT | STATEMENT NUMBER | CONTINUATION | FORTRAN STATEMENT | IDENTIFICATION
0 0 0 0 0 | 0 | 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 | 73 74 75 76 77 78 79 80
1 2 3 4 5 | 6
```

| | | | |
|---|---|---|---|
| A. | F5.0 | SUMCL(J,1) | MINE TYPE NUMBER |
| B. | F5.2 | SUMCL(J,2) | PROBABILITY OF IMPACT DAMAGE ON THE HARDEST BOTTOM FOR THIS MINE TYPE. REFLECTS THE LIKLIHOOD MINE RETAINS SOME PROPULSIVE VELOCITY ON STRIKING BOTTOM. |
| C. | F5.2 | SUMCL(J,3) | PROBABILITY OF BURYING IN SOFTEST BOTTOM FOR BOTTOM MINES OF THIS TYPE. INSERT ZERO FOR MOORED MINES. |
| D. | F3.0 | SUMCL(J,4) | INDICATOR OF MOOR OR BOTTOM TYPE. SEE ITEM J OF ACMCL. |
| E. | F5.0 | SUMCL(J,5) | MINE TYPE I,(NWL TYPE), KEY TO ACTUATION CURVES FOR MINES BURYING OR NOT BURYING WHICHEVER IS UNDESIRABLE FOR BOTTOM MINES OF THIS TYPE AND SHOULD REFLECT A DIMINISHED CAPABILITY. INSERT ZERO FOR MOORED MINES. |
| F. | F5.2 | SUMCL(J,6) | PROBABILITY OF CASE AND ANCHOR SEPARATION FOR MOORED MINES OF THIS TYPE. INSERT ZERO FOR BOTTOM MINES. |
| G. | F5.2 | SUMCL(J,7) | PROBABILITY OF FULL ANCHOR CABLE DEPLOYMENT FOR MOORED MINES OF THIS TYPE. INSERT ZERO FOR BOTTOM MINES. |
| H. | F5.2 | SUMCL(J,8) | PROBABILITY OF A NORMAL RUN FOR A SELF-PROPELLED MINE OF THIS TYPE. INSERT ZERO FOR TYPES NOT SELF-PROPELLED. |
| I. | F5.2 | SUMCL(J,9) | PROBABILITY OF MOTOR FAILURE GIVEN A FAILURE FOR SELF-PROPELLED MINES OF THIS TYPE. INSERT ZERO FOR TYPES NOT SELF-PROPELLED. |

| | | | |
|---|---|---|---|
| J. | F5.2 | SUMCL(J,10) | PROBABILITY OF GYRO FAILURE GIVEN A FAILURE FOR SELF-PROPELLED MINES OF THIS TYPE. INSERT ZERO FOR TYPES NOT SELF-PROPELLED. |
| K. | F5.2 | SUMCL(J,11) | PROBABILITY OF DEPTH CONTROL FAILURE GIVEN A FAILURE FOR SELF-PROPELLED MINES OF THIS TYPE. INSERT ZERO FOR TYPES NOT SELF-PROPELLED. |
| L. | F5.0 | SUMCL(J,12) | CEP IN YARDS FOR A NORMAL RUN OF A SELF-PROPELLED MINE OF THIS TYPE. INSERT ZERO FOR TYPES NOT SELF-PROPELLED. |
| M. | F5.2 | | PROBABILITY THAT THE MINE WILL OPERATE GIVEN IT HAS NOT BEEN DAMAGED, I.E., THE MINES RELIABILITY. |

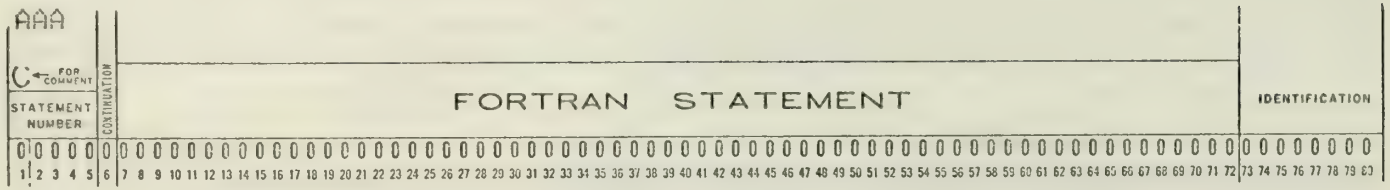
CARD NO. THIRTEEN
NUMBER OF CARDS REQUIRED = THIRTEEN
ARRAY NAME = MATRIX OF RANDOM NUMBER GENERATOR INITIALIZING VALUES (IRMAT, 100)




| | | | |
|---|---|---|---|
| A. THRU H. | I10 | IRMAT(J) | RANDOM NUMBER GENERATOR INITIALIZING VALUE FOR EACH ITERATION. |

NOTE- THIRTEENTH CARD ONLY HAS ITEMS A. THRU D.

CARD NO. FOURTEEN
NUMBER OF CARDS REQUIRED = ONE




A. I3 NREP NUMBER OF REPETITIONS DESIRED.
NOTE - NREP CAN NOT EXCEED 100 WITHOUT INCREASING THE SIZE OF IRMAT.

VARIABLE NAMES

THOSE VARIABLE NAMES WHICH REPRESENT INPUT DATA ARE DISTINGUISED BY ASTERISKS.

ALPHA = LOGF(PU+1.)/.7,USED IN CALCULATION OF PROB OF DAMAGE FOR SHIPS.
ANE = AVERAGE NAVIGATIONAL ERROR (SHIPS AND SUBMARINES). (*)
BNE = BEST POSSIBLE NAVIGATIONAL ERROR (SHIPS AND AIRCRAFT). (*)
BTMTYP= BOTTOM TYPE VALUE.
CEP = CIRCULAR PROBABLE ERROR USED FOR AIRCRAFT DROPPED MINES AND SELF -PROPELLED SUBMARINE LAYED MINES. (*)
DELAY = TIME DELAY VALUE USED FOR SHIP AND SUBMARINE MINE LAYING.
DRUN = TOTAL RUN OF A SUBMARINE SELF-PROPELLED MINE WHICH SUFFERS A DEPTH FAILURE
DT = VALUE OF DEAD(ATTRITION) TIME FOR AIRCRAFT, DAMAGE TIME FOR SHIPS,OR DETECTION EVENT TIME FOR SUBMARINES.
DX = X-COMPONENT OF THE RANGE ERROR IN CALCULATION OF XPOSIT.
DY = Y-COMPONENT OF THE RANGE ERROR IN CALCULATION OF YPOSIT.
ECCENT= ECCENTRICITY OF FLIGHT PATH OF AIRCRAFT DROPPED MINES.
GNE = GREATEST NAVIGATIONAL ERROR (SHIPS AND SUBMARINES). (*)
GYPHI = RANDOM COURSE ERROR OF A SUBMARINE SELF-PROPELLED MINE WHICH HAS SUFFERED A GYRO FAILURE.
GYRUN = RANDOM RUN OF A SUBMARINE SELF-PROPELLED MINE WHICH HAS SUFFERED A GYRO FAILURE.
HYP = TOTAL RANGE ERROR FROM MINE INTENDED POSITION.
IABC = DUMMY INDEX USED IN VEHICLE SELECTION LOOP.
IDAM = DAMAGE TYPE FLAG(SHIPS)
IDET = DETECTION EVENT FLAG(SUBMARINES)
IDET1 = DETECTION EVENT FLAG(SUBMARINES)
IDOT = INDEX VALUE USED IN SUBMARINE MINES REMAINING ON BOARD STORE EVENT.
II = INDEX VALUE USED IN TIME BETWEEN POINTS ON A LEG LOOP.
III = INDEX VALUE USED IN INITIALIZING OUTPUT MATRIX.
IL = INDEX VALUE USED IN LEG NUMBER DECODING EVENT.
IR = INTEGER IDENTIFIER FOR RAN1 ROUTINE. (*)
IREP = THE NUMBER OF THE REPETITION BEING PLAYED (INDEX VALUE)

ISUB = INDEX VALUE USED IN SUBMARINE TAM DETERMINING EVENT.
I1,I2,= DUMMY VARIABLES USED IN THE CALCULATION OF TIME BETWEEN POINTS
I3,I4 ON A LEG.
J = VEHICLE TYPE DESIGNATOR.
JCIG = INDEX VALUE USED IN SUBMARINE DETECTION EVENT LOOP.
JI1 = INDEX USED IN SUBMARINE LEG DATA TEMPORARY STORAGE LOOP.
JI2 = INDEX USED IN SUBMARINE LEG DATA TEMPORARY STORAGE LOOP.
JIK = INDEX USED IN SUBMARINE LEG DATA TEMPORARY STORAGE LOOP.
JJ = VEHICLE TYPE IDENTIFIER.
JU = INDEX USED IN SHIP TIME INCREMENT CORRECTION EVENT.
JUP = INDEX USED IN SUBMARINE TIME INCREMENT CORRECTION EVENT.
K = INDEX USED IN LEG LOOP
KK = INDEX USED IN CALCULATION OF EVENT TIMES LOOP.
KKK = INDEX USED IN SHIP AND AIRCRAFT LEG DATA TEMPORARY STORAGE LOOPS
L = LEG NUMBER.
LDFLAG= SUBMARINE DETECTION EVENT FLAG.
LL1,2,
3AND4 = INDEXES USED FOR TEMPORARY STORAGE OF BOMBRACK MATRIX LOOPS.
MACT = NUMBER OF AIRCRAFT MINE TYPES. (*)
MFLAG = MOOR OR BOTTOM TYPE INDICATOR.
MINTOT= TOTAL NUMBER OF MINES TO BE LAYED. (*)
MLFLAG= FLAG INDICATING WHEN OUTPUT MATRIX IS TO BE WRITTEN ON TAPE.
MM = LEG EVENT INDICATOR.
MMAX = DEGREE OF DEPTH CONTOUR POLYNOMIAL IN Y. (*(TAPE))
MMAX1 = DEGREE OF BOTTOM CONTOUR POLYNOMIAL IN Y (*(TAPE))
MQ = INDEX TO ELEMENT OF TD ARRAY CONTAINING TOTAL TIME.
MR = COUNTER FOR SUBMARINE MINES REMAINING ON BOARD.
MSHT = NUMBER OF SHIP MINE TYPES. (*)
MSUT = NUMBER OF SUBMARINE MINE TYPES. (*)
MT2 = MINE TYPE.
M1 = MINE NUMBER.
NA = NUMBER OF AIRCRAFT (*)
NACC = NUMBER OF COLUMNS IN ACLDL. (*)
NACR = NUMBER OF ROWS IN ACLDL. (*)
NAT = NUMBER OF AIRCRAFT TYPES. (*)

```
NBR    = NUMBER OF BOMBRACKS PER AIRCRAFT.
NDAM   = SHIP DAMAGE TYPE INDICATOR.
NIL    = IL+5, DUMMY SUBSCRIPT USED IN LEG NUMBER DECODING EVENT.
NJ1,
NJ11,  = DUMMY VARIABLES USED IN SUBMARINE DATA TRANSFER EVENT.
NJ2,
NJ22
NLEGR  = NUMBER OF LEGS REMAINING FOR A VEHICLE.
NM     = INDEX TO MINE NUMBER IN ANY VEHICLE DESCRIPTION LIST.
NMAX   = DEGREE OF DEPTH CONTOUR POLYNOMIAL IN X. (*(TAPE))
NMAX1  = DEGREE OF BOTTOM CONTOUR POLYNOMIAL IN X. (*(TAPE))
NMINES= NUMBER OF MINES TO BE LAYED ON A LEG.
NMIN   = INDEX FOR MINE LAYING LOOP.
NMINR  = INDEX FOR NUMBER OF MINES REMAINING LOOP IN ATTRITION EVENTS.
NREP   = NUMBER OF REPITIONS OF THE GAME, EACH REPITION USING A DIFFERENT
         VALUE OF IR.  (*)
NSH    = TOTAL NUMBER OF SHIPS. (*)
NSHC   = NUMBER OF COLUMNS IN SHLDL. (*)
NSHR   = NUMBER OF ROWS IN SHLDL. (*)
NSHT   = TOTAL NUMBER OF SHIP TYPES. (*)
NSU    = TOTAL NUMBER OF SUBMARINES. (*)
NSUC   = NUMBER OF COLUMNS IN SULDL. (*)
NSUR   = NUMBER OF ROWS IN SULDL. (*)
NSUT   = TOTAL NUMBER OF SUBMARINE TYPES. (*)
NTOTLG= TOTAL NUMBER OF LEGS ASSIGNED TO A VEHICLE. (*)
NUBR   = INDEX FOR AIRCRAFT MINE RELEASE EVENT.
NUBRS  = NUMBER OF BOMB RACKS FOR AN AIRCRAFT. (*)
NV     = TOTAL NUMBER OF VEHICLES THAT WILL TRANSIT MINEFIELD. (*)
NVEH   = VEHICLE NUMBER. (*)
N1     = NUMBER OF COLUMNS OF A LEG DESCRIPTION LIST USED FOR A LEG, AND
         INDEX FOR LEG DESCRIPTION LIST DATA TRANSFER EVENT.
N      = NUMBER OF TIME INCREMENTS ON A LEG.
PB     = PROBABILITY OF BURYING.
PD     = PROBABILITY OF A MINE BEING DAMAGED ON BOTTOM IMPACT.
```

PDAM = PROBABILITY OF SHIP SUFFERING DAMAGE.
PDHB = PROBABILITY OF A MINE BEING DAMAGED ON THE HARDEST BOTTOM. (*)
PU = PROBABILITY OF A SHIP SUFFERING SOME DAMAGE DURING ITS FIRST HALF-HOUR IN THE MINEFIELD. (*)
RN = A RANDOM (UNIFORM) NUMBER.
RUN = DISTANCE TRAVELED BY A SUBMARINE FIRED SELF-PROPELLED MINE.
S = VEHICLE MINE LAYING SPEED (YARDS/MIN).
SDELAY= TIME DELAY IN FIRING A SUBMARINE MINE.
SIGMA = CEP/1.1744
T = TIME PERIOD FOR WHICH THE PROBABILITY OF A SHIP SUFFERING DAMAGE BECOMES UNITY.
TAM = TIME AT WHICH SUBMARINE MUST CEASE OPERATIONS IN MINEFIELD. (*)
TDIF = TOTAL TIME(IN MINUTES) THAT A SUBMARINE HAS LEFT TO OPERATE IN THE MINEFIELD.
TDT = TRUE EVENT TIME.
TDUM = ADJUSTED MINE RELEASE TIME (SHIP).
TD100 = TRANSIT TIME FOR SUBMARINE FROM THE MINEFIELD TO THE 100 FATHOM CURVE.
TE = INCREMENTAL TIME BETWEEN MINE INTENDED RELEASE POSITIONS ON A LEG.
TIP = TRUE TIME OF A VEHICLE AT ITS INITIAL POSITION FOR A LEG.
TLAY = TRUE TIME OF LAY OF A MINE.
TOTTIM= TOTAL TIME A VEHICLE WILL REMAIN IN MINEFIELD.
TPX = X COORDINATE OF A VEHICLE TERMINAL POINT.
TPY = Y COORDINATE OF A VEHICLE TERMINAL POINT.
TQ = DIFFERENCE BETWEEN TDIF AND TD100 (TIME A SUBMARINE WILL HAVE LEFT TO OPERATE AFTER TRANSITING FROM THE 100 FATHOM CURVE TO THE MINEFIELD.
TREF = ADJUSTED EVENT TIME FOR SUBMARINES.
TREF1,= ADJUSTED EVENT TIMES FOR SHIPS.
TREF2
TREM = TIME REMAINING (FROM TOTTIM) FOR SUBMARINES.
TRET = TIME A SUBMARINE RETURNS TO THE MINEFIELD.
TTP = TRUE TIME OF ARRIVAL OF A VEHICLE AT A TERMINAL POINT.
TTRET = TRUE TIME A SUBMARINE RETURNS TO THE MINEFIELD.

```
WD     = WATER DEPTH(FT)
XDELE  = DISTANCE TRAVELED IN THE X DIRECTION BY A SHIP DURING DELAY IN
         MINE RELEASE.
XE     = X ERROR IN FINAL MINE POSITION DUE TO NAVIGATIONAL ERROR OR CEP.
XIPX   = X COORDINATE OF A VEHICLE INITIAL POSITION.
XMEANX= MIDPOINT (X-AXIS) OF THE GRIDED RECTANGLE WHICH SURROUNDS THE
         MINEFIELD (SEE SURFIT DESCRIPTION). (*(TAPE))
XPOSIT= FINAL X COORDINATE OF A MINE.
XTRACL
XTRACR= UNPACKED PORTIONS OF CODED BOMB RACK WORD.
XTRACX
YDELE  = DISTANCE TRAVELED IN THE Y DIRECTION BY A SHIP DURING DELAY IN
         MINE RELEASE.
YE     = Y ERROR IN FINAL MINE POSITION DUE TO NAVIGATIONAL ERROR OR CEP.
YIPY   = Y COORDINATE OF A VEHICLE INITIAL POSITION.
YMEANY= MIDPOINT (Y-AXIS) OF THE GRIDED RECTANGLE WHICH SURROUNDS THE
         MINEFIELD (SEE SURFIT DESCRIPTION). (*(TAPE))
YPOSIT= FINAL X COORDINATE OF A MINE.


ARRAY NAMES

ACCL   = AIRCRAFT CHARACTERISTICS LIST
ACLDL  = AIRCRAFT LEG DESCRIPTION LIST
ACMCL  = AIRCRAFT MINE CHARACTERISTICS LIST
BOT    = MATRIX OF COEFFICIENTS OF BOTTOM-TYPE POLYNOMIAL
BR     = AIRCRAFT BOMB RACK LIST
BRTEMP= TEMPORARY STORAGE FOR BR
FXPTS  = LIST OF X COORDINATES OF POINTS DEFINING THE MINEFIELD
FYPTS  = LIST OF Y COORDINATES OF POINTS DEFINING THE MINEFIELD
IRMAT  = MATRIX OF IR VALUES
MDL    = MINE DESCRIPTION LIST
OUTPUT= MATRIX OF OUTPUT DATA FOR EACH MINE LAYED.
PHI    = MATRIX OF COEFFICIENTS OF THE DEPTH CONTOUR POLYNOMIAL
```

PROB = LIST OF PROBABILITIES OF SUBMARINE DETECTION EVENTS.
SHCL = SHIP CHARACTERISTIC LIST
SHLDL = SHIP LEG DESCRIPTION LIST
SHMCL = SHIP MINE CHARACTERISTIC LIST
SMROB = MATRIX OF MINES REMAINING ON BOARD A SUBMARINE THAT IS ATTRITED OR ABORTS ITS MISSION
SUCL = SUBMARINE CHARACTERISTIC LIST
SULDL = SUBMARINE LEG DESCRIPTION LIST
SUMCL = SUBMARINE MINE CHARACTERISTIC LIST
TD. = MATRIX OF RELATIVE EVENT TIMES
VAL = VEHICAL ARRIVAL LIST
XLCOD = MATRIX OF CODED LEG ASSIGNMENTS
XLDL = TEMPORARY STORAGE FOR VEHICLE LEG DATA

# APPENDIX V

## FLOWCHARTS FOR PROGRAM MINDEL

The succeeding pages of this appendix contain the flowcharts of the logic of program MINDEL and its subroutines. A list of the flowchart symbols used is shown in Figure 18. In addition to the normal application of these symbols, i.e., an appropriate symbol containing a FORTRAN statement, the processing rectangle containing a brief descriptive statement and with a number, e.g., 1. or 3.2, over the upper left-hand corner is used to summarize certain sections of the program as an aid in following the logic. A later flowchart with the same number appearing in a small box at the upper right-hand corner of the page and delineated by the terminal symbol with the words, "ENTRY" and "CONTINUE", contains further detail of the section summarized. Also, the connector numbers are consistent throughout the entire set of flowcharts with exception of the subroutine flowcharts. These flowcharts will greatly aid the user in understanding the commentary in the main body of this thesis as well as the program structure.

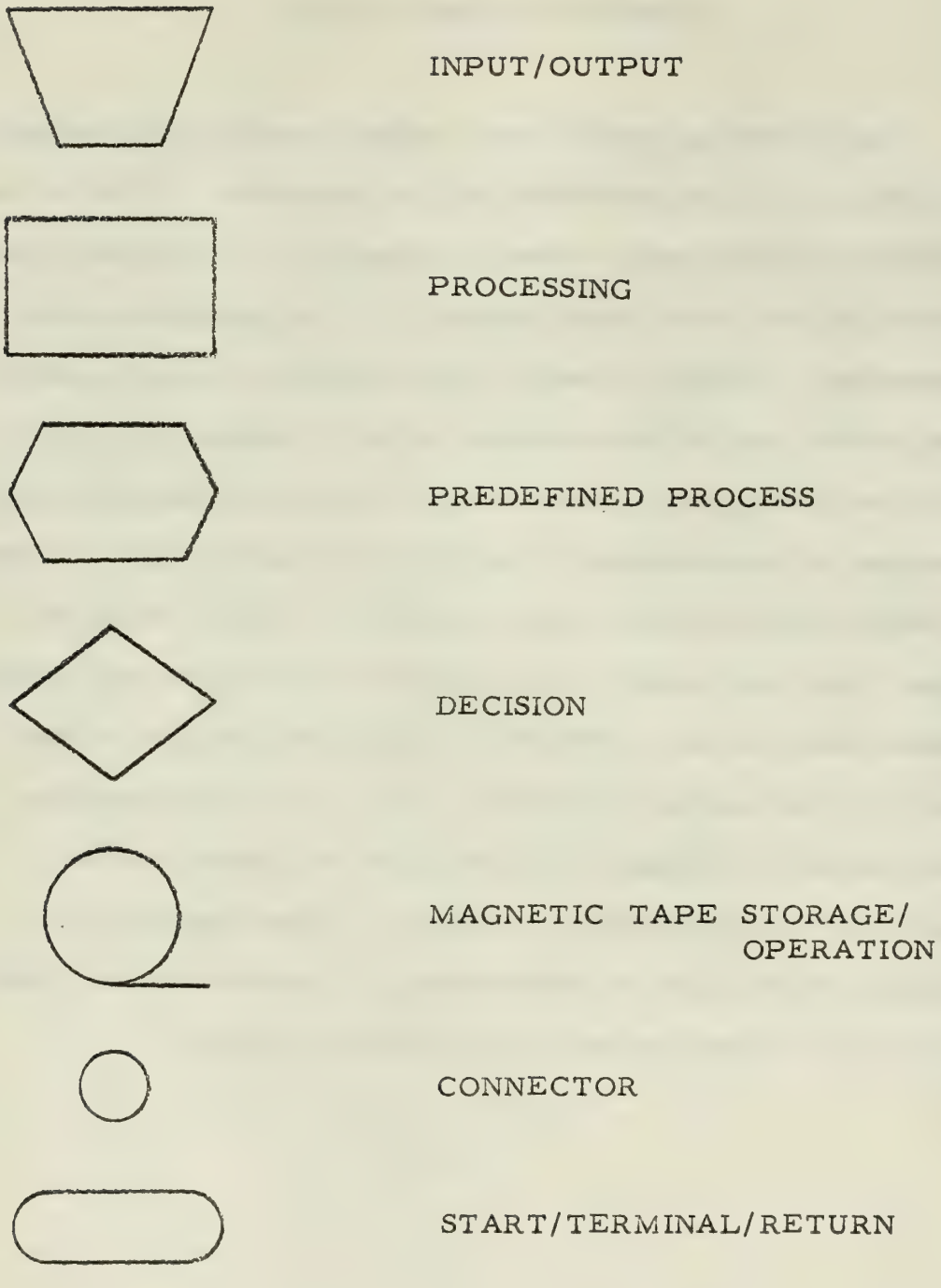
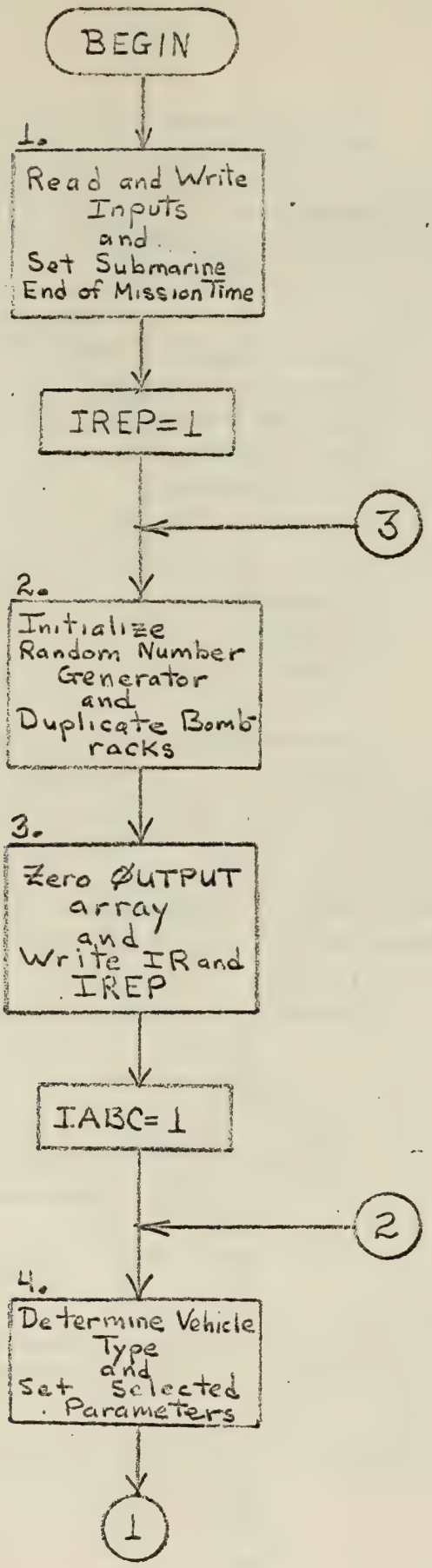
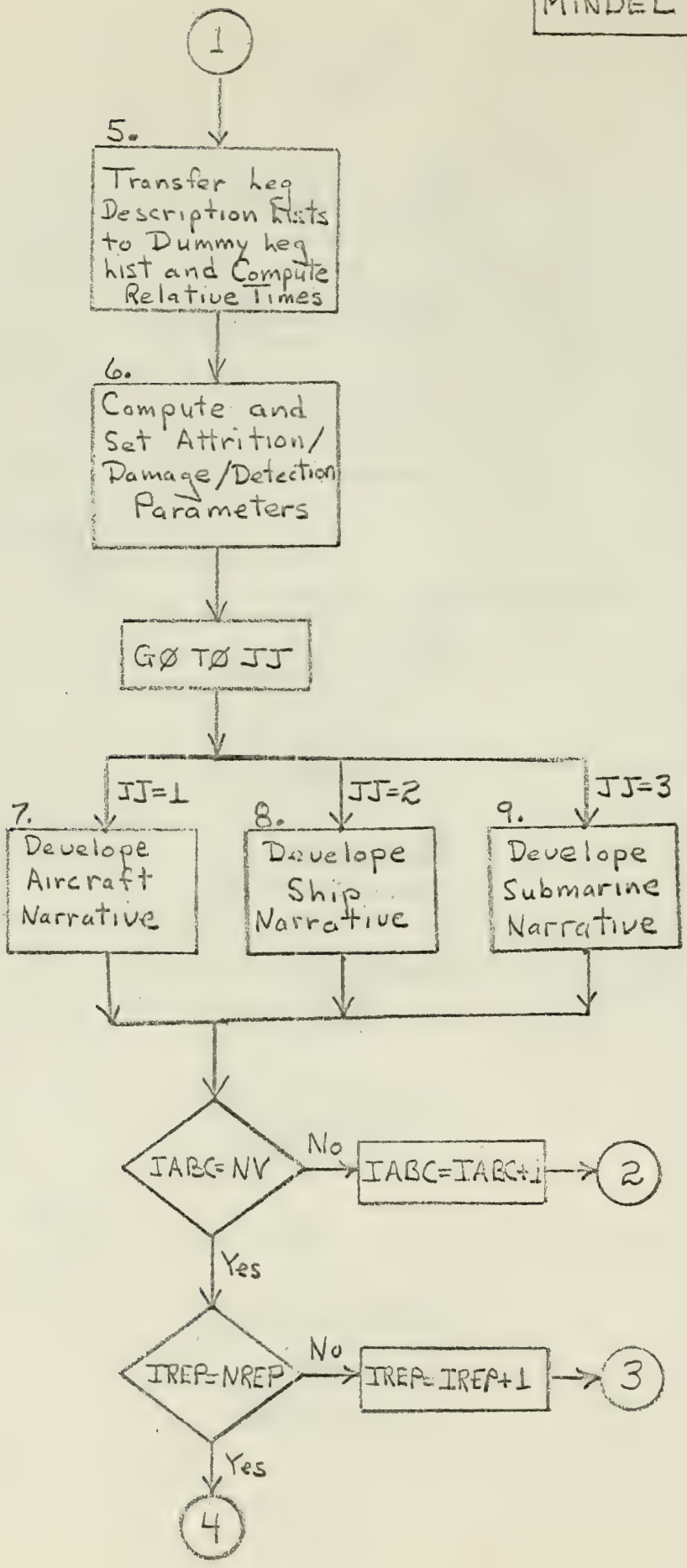
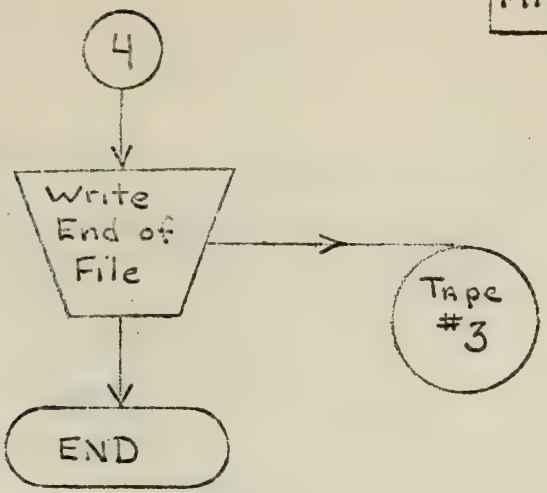


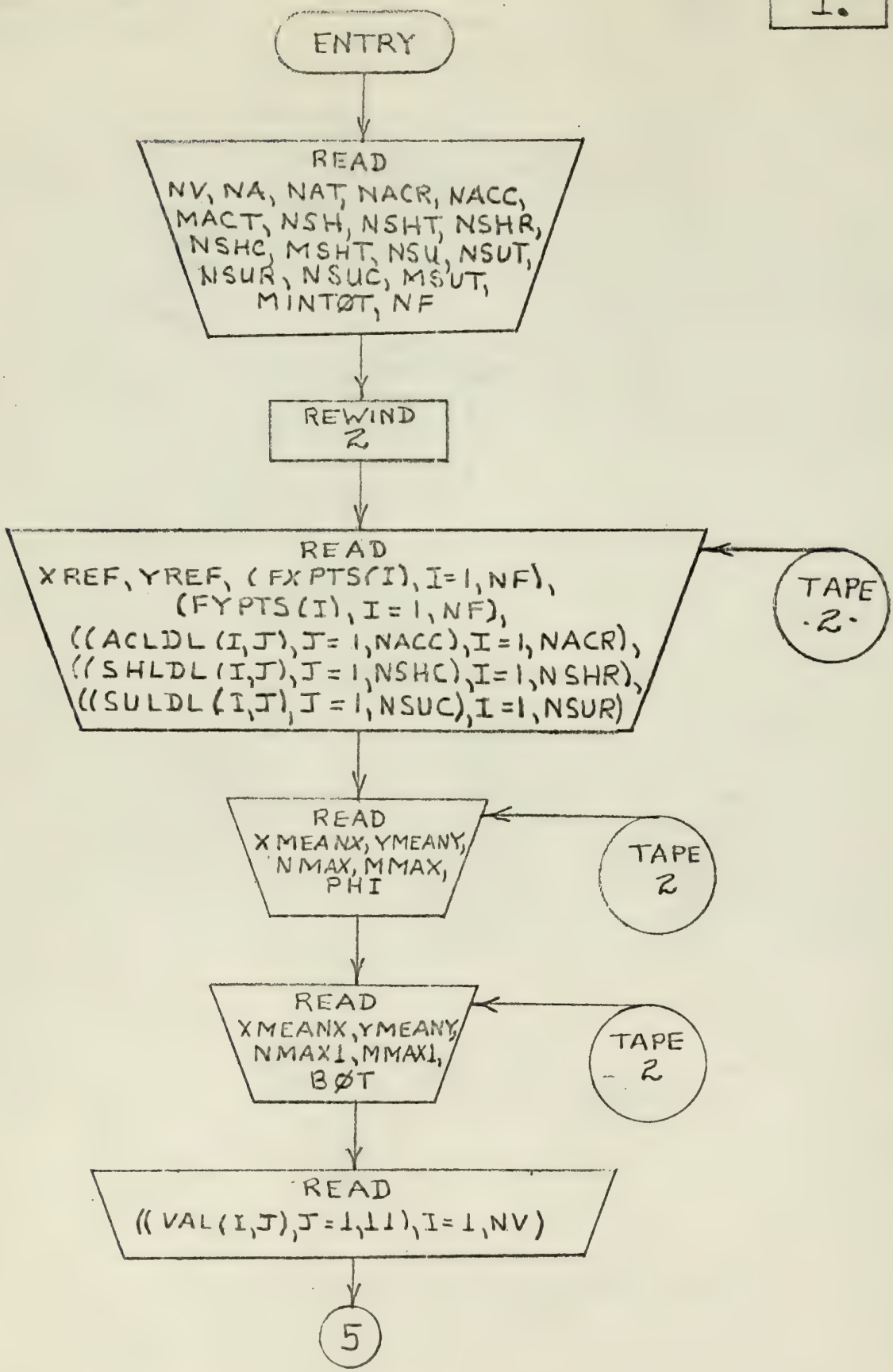

Figure 18. Flowchart Symbols

# Program MINDEL

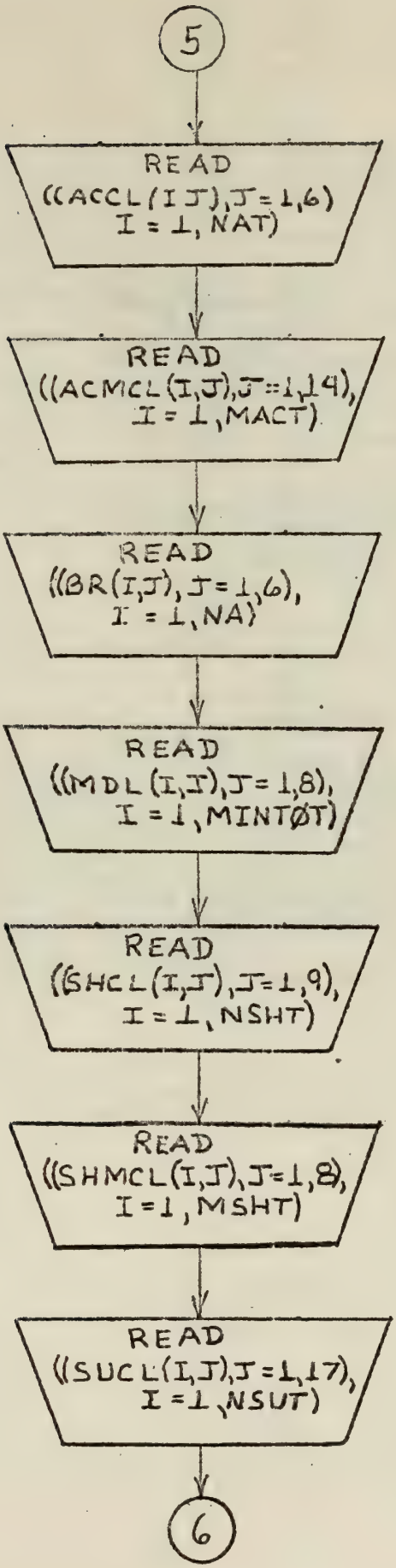



1
5.
Transfer Neg Description Mats to Dummy Neg Mat and Compute Relative Times
6.
Compute and Set Attrition/ Damage/Detection Parameters
GØ TØ JJ
JJ=1
JJ=2
JJ=3
7.
Develope Aircraft Narrative
8.
Develope Ship Narrative
9.
Develope Submarine Narrative
IABC=NV
No
IABC=IABC+1
2
Yes
IREP=NREP
No
IREP=IREP+1
3
Yes
4

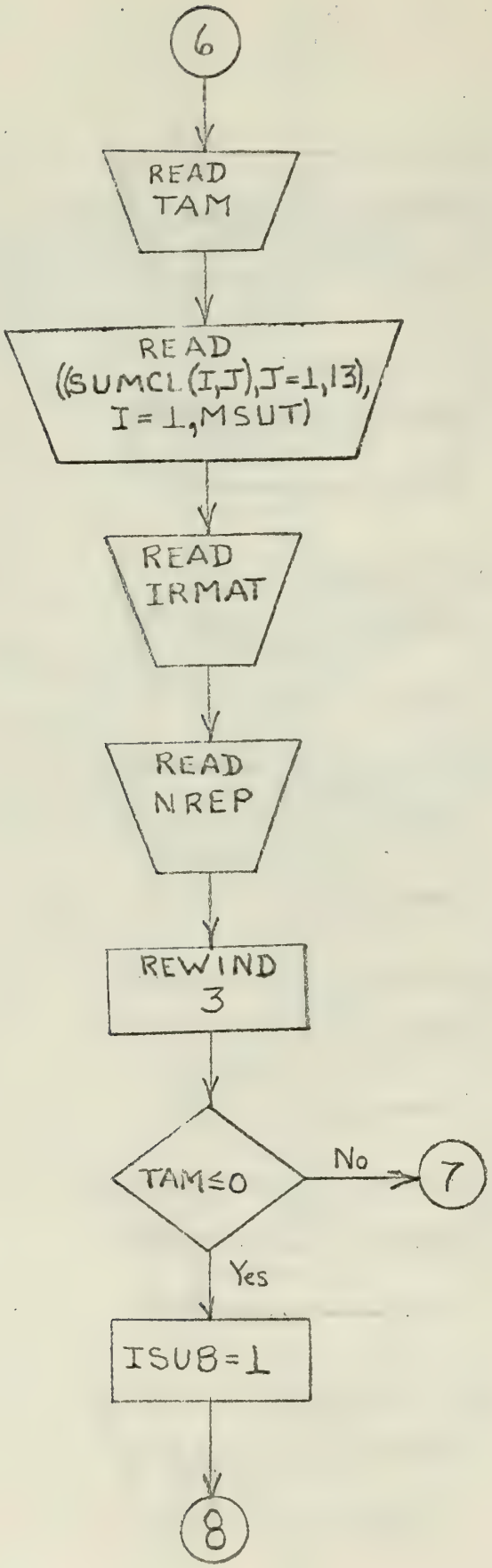

4
Write End of File
Tape #3
END


ENTRY
READ
NV, NA, NAT, NACR, NACC,
MACT, NSH, NSHT, NSHR,
NSHC, MSHT, NSU, NSUT,
NSUR, NSUC, MSUT,
MINTØT, NF
REWIND
2
READ
XREF, YREF, (FXPTS(I), I=1, NF),
(FYPTS(I), I=1, NF),
((ACLDL(I,J), J=1, NACC), I=1, NACR),
((SHLDL(I,J), J=1, NSHC), I=1, NSHR),
((SULDL(I,J), J=1, NSUC), I=1, NSUR)
TAPE
2
READ
XMEANX, YMEANY,
NMAX, MMAX,
PHI
TAPE
2
READ
XMEANX, YMEANY,
NMAX1, MMAX1,
BØT
TAPE
2
READ
((VAL(I,J), J=1,11), I=1, NV)
5

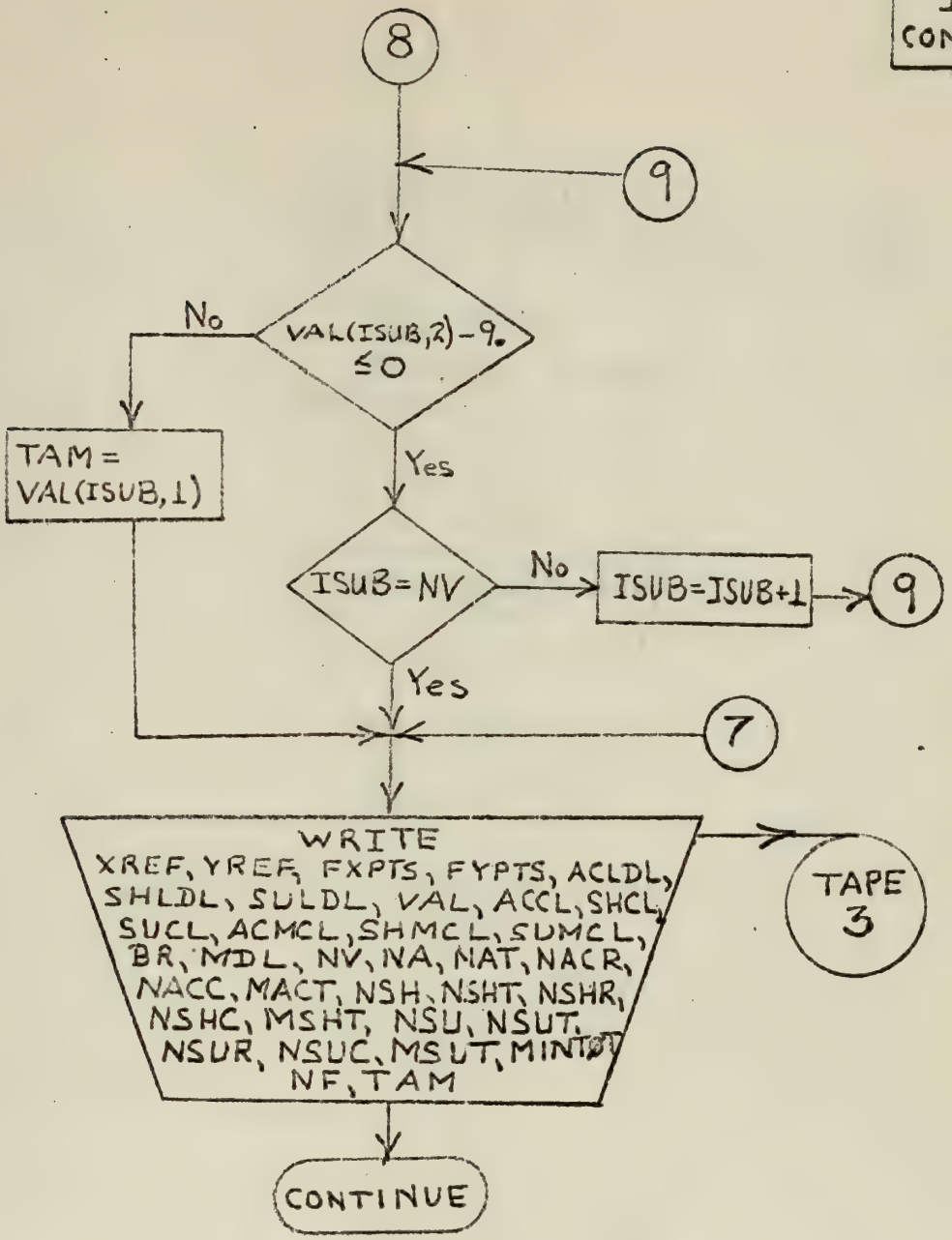
1. CONTD

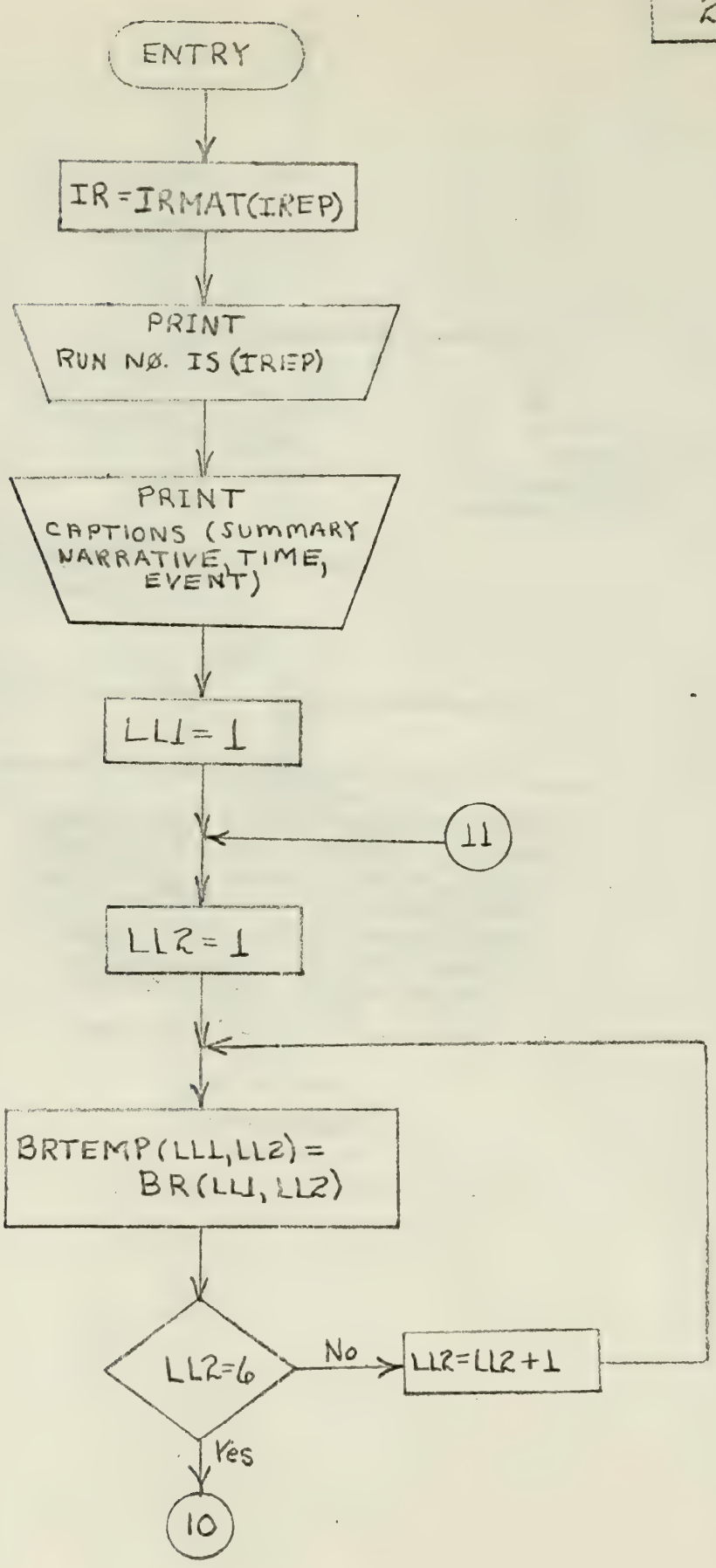

5
READ ((ACCL(IJ),J=1,6) I=1,NAT)
READ ((ACMCL(I,J),J=1,19), I=1,MACT)
READ ((BR(I,J),J=1,6), I=1,NA)
READ ((MDL(I,J),J=1,8), I=1,MINTØT)
READ ((SHCL(I,J),J=1,9), I=1,NSHT)
READ ((SHMCL(I,J),J=1,8), I=1,MSHT)
READ ((SUCL(I,J),J=1,17), I=1,NSUT)
6

1. CONT'D

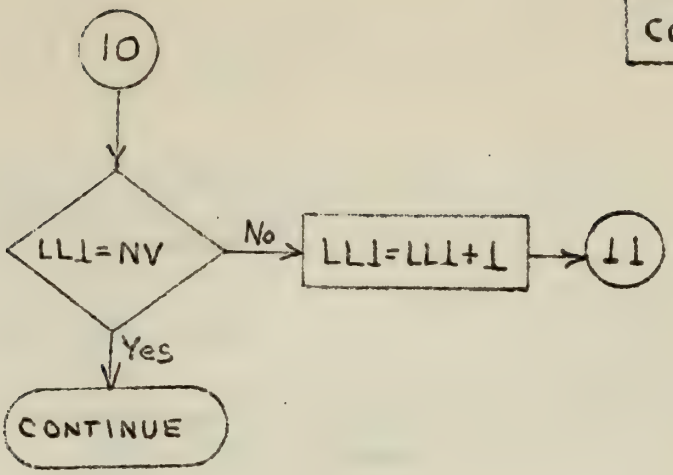

6
READ TAM
READ ((SUMCL(I,J),J=1,13), I=1,MSUT)
READ IRMAT
READ NREP
REWIND 3
TAM≤0
No
7
Yes
ISUB=1
8

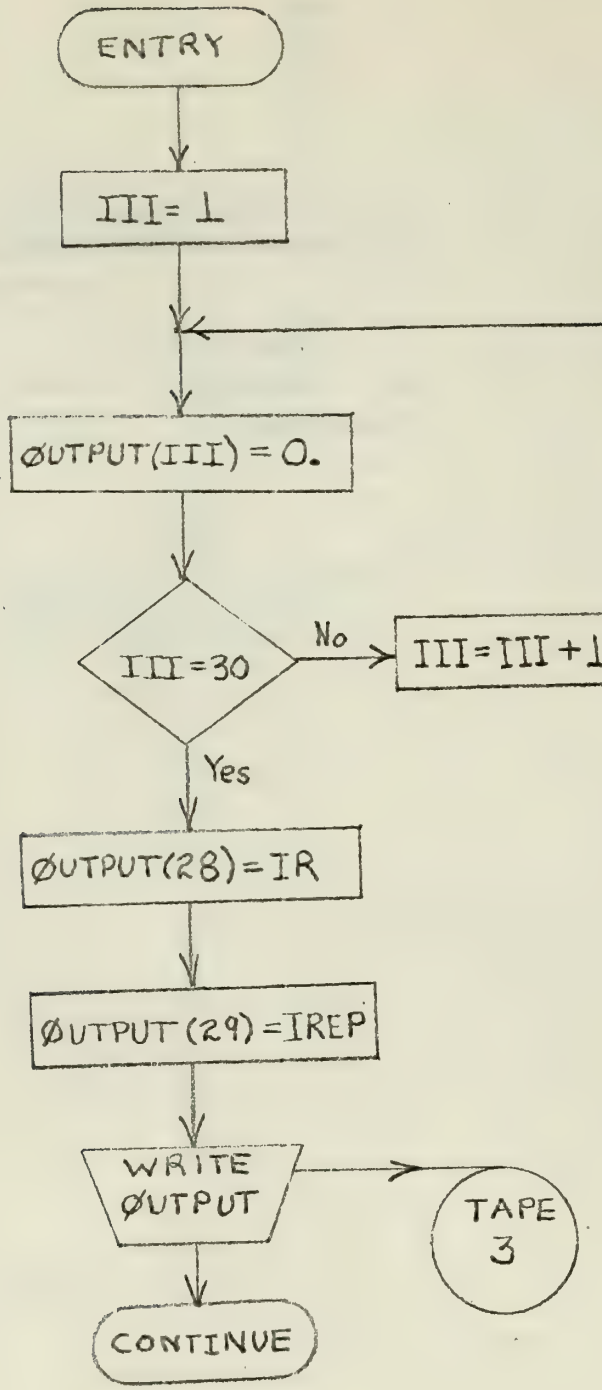
1. CONTD

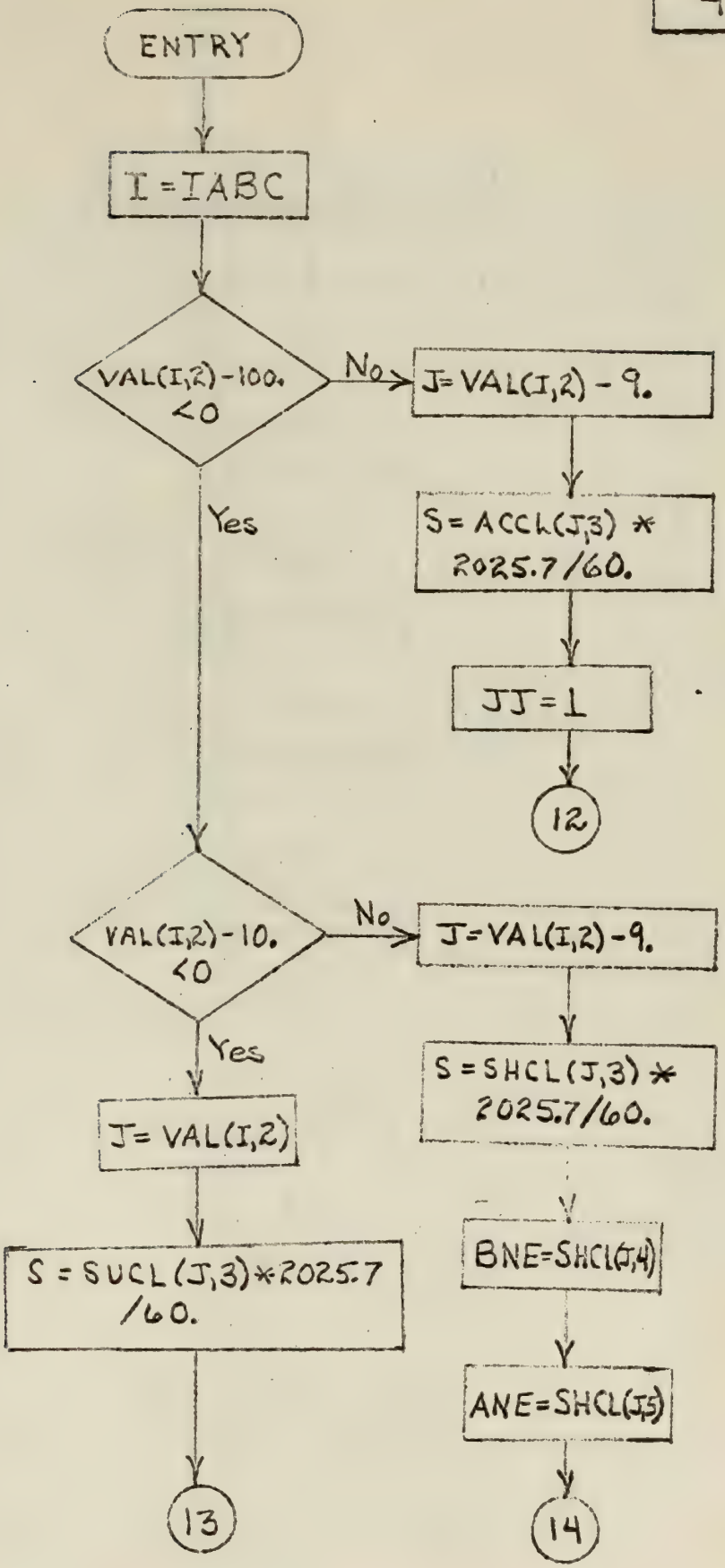

8
9
No
VAL(ISUB,2)-9. ≤0
TAM = VAL(ISUB,1)
Yes
ISUB=NV
No
ISUB=ISUB+1
9
Yes
7
WRITE
XREF, YREF, FXPTS, FYPTS, ACLDL, SHLDL, SULDL, VAL, ACCL, SHCL, SUCL, ACMCL, SHMCL, SUMCL, BR, MDL, NV, NA, MAT, NACR, NACC, MACT, NSH, NSHT, NSHR, NSHC, MSHT, NSU, NSUT, NSUR, NSUC, MSUT, MINT, NF, TAM
TAPE 3
CONTINUE

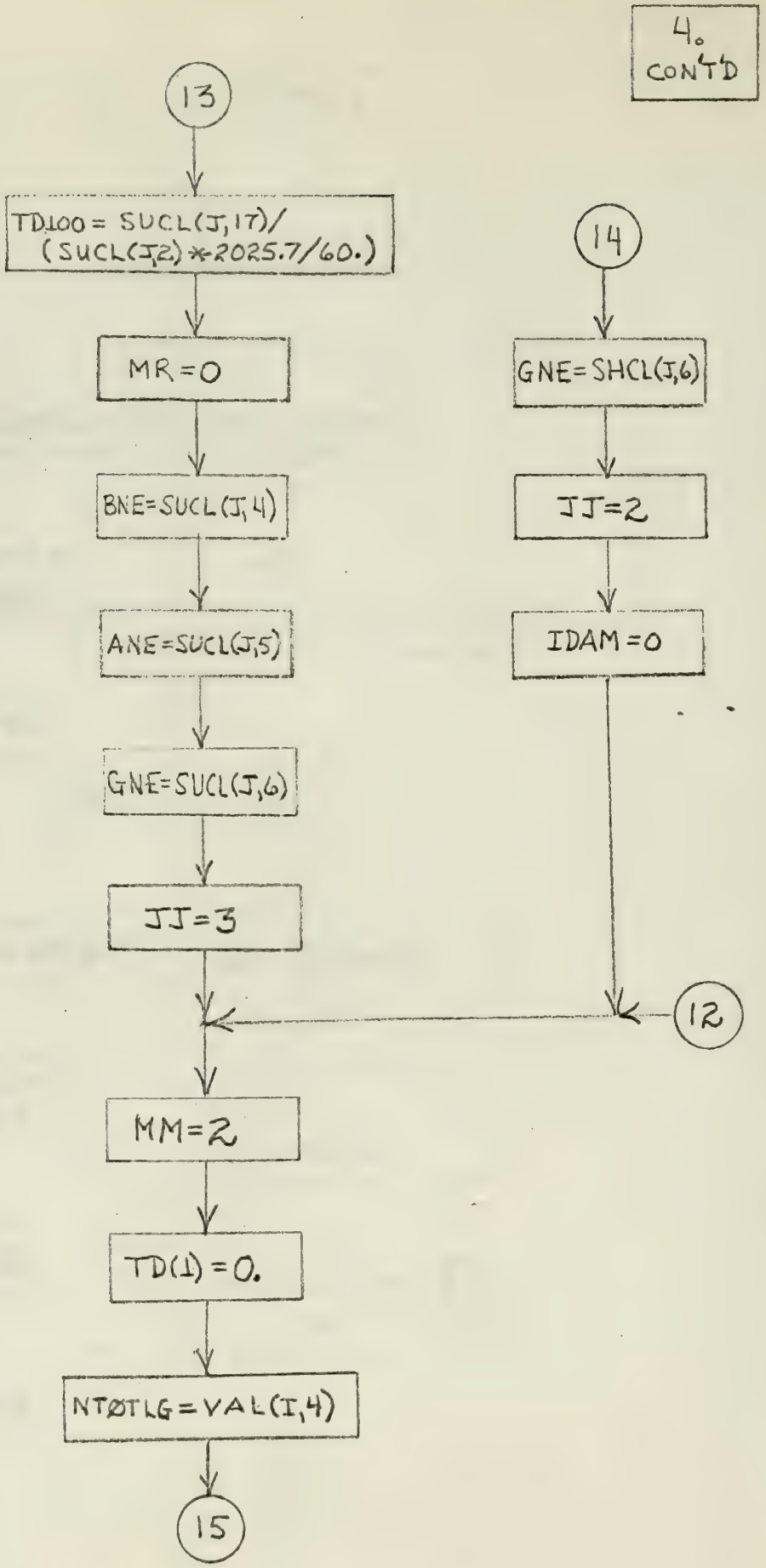

ENTRY
IR=IRMAT(IREP)
PRINT
RUN NØ. IS (IREP)
PRINT
CAPTIONS (SUMMARY NARRATIVE, TIME, EVENT)
LL1=1
11
LL2=1
BRTEMP(LL1,LL2)=
BR(LL1,LL2)
LL2=6
No
LL2=LL2+1
Yes
10

2. CONTD

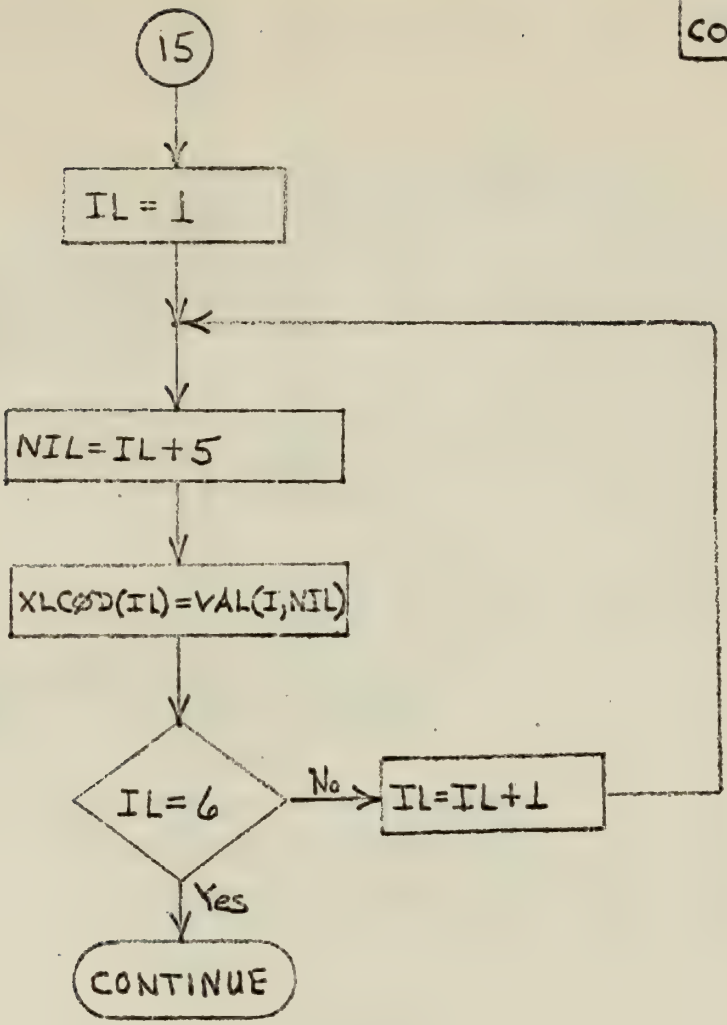


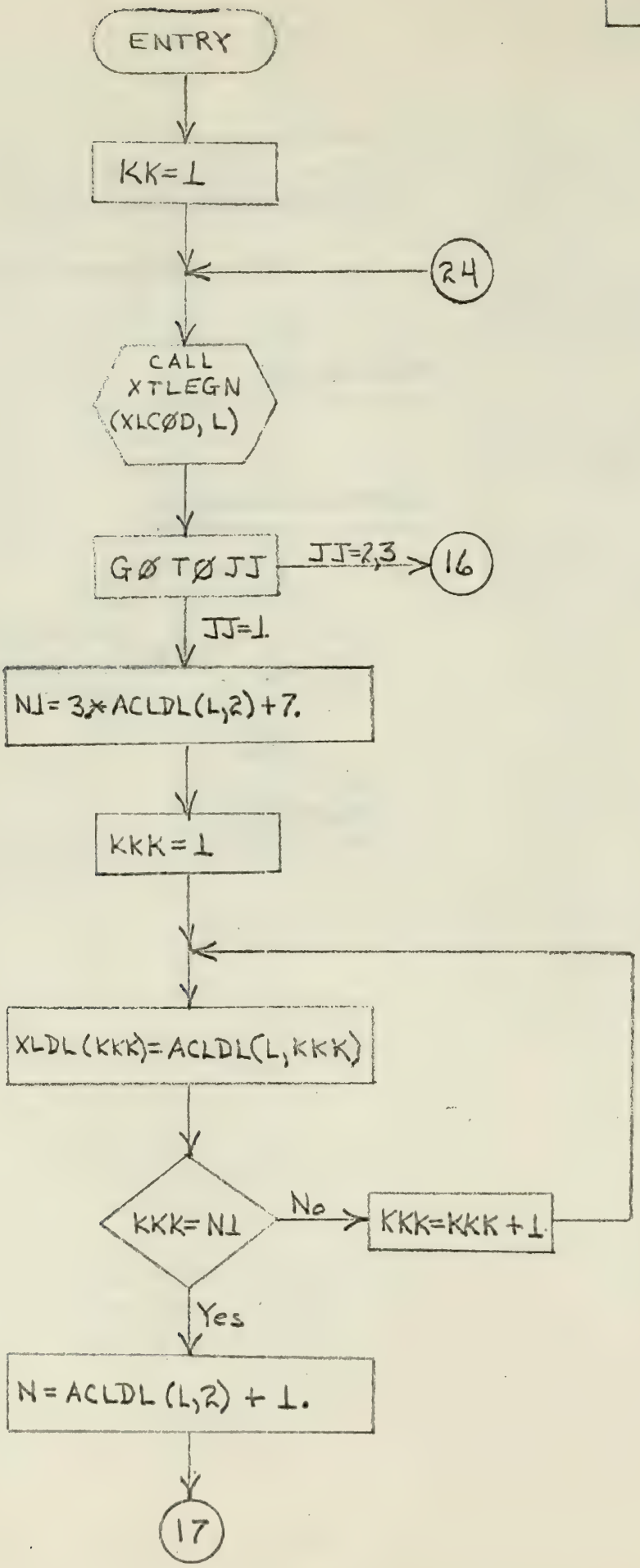

ENTRY
III= 1
ØUTPUT(III) = 0.
III=30
No
III=III+1
Yes
ØUTPUT(28)=IR
ØUTPUT(29)=IREP
WRITE ØUTPUT
TAPE 3
CONTINUE


ENTRY
I = IABC
VAL(I,2) - 100. < 0
No
J = VAL(I,2) - 9.
S = ACCL(J,3) * 2025.7/60.
JJ = 1
12
Yes
VAL(I,2) - 10. < 0
No
J = VAL(I,2) - 9.
S = SHCL(J,3) * 2025.7/60.
BNE = SHCL(J,4)
ANE = SHCL(J,5)
14
Yes
J = VAL(I,2)
S = SUCL(J,3) * 2025.7/60.
13

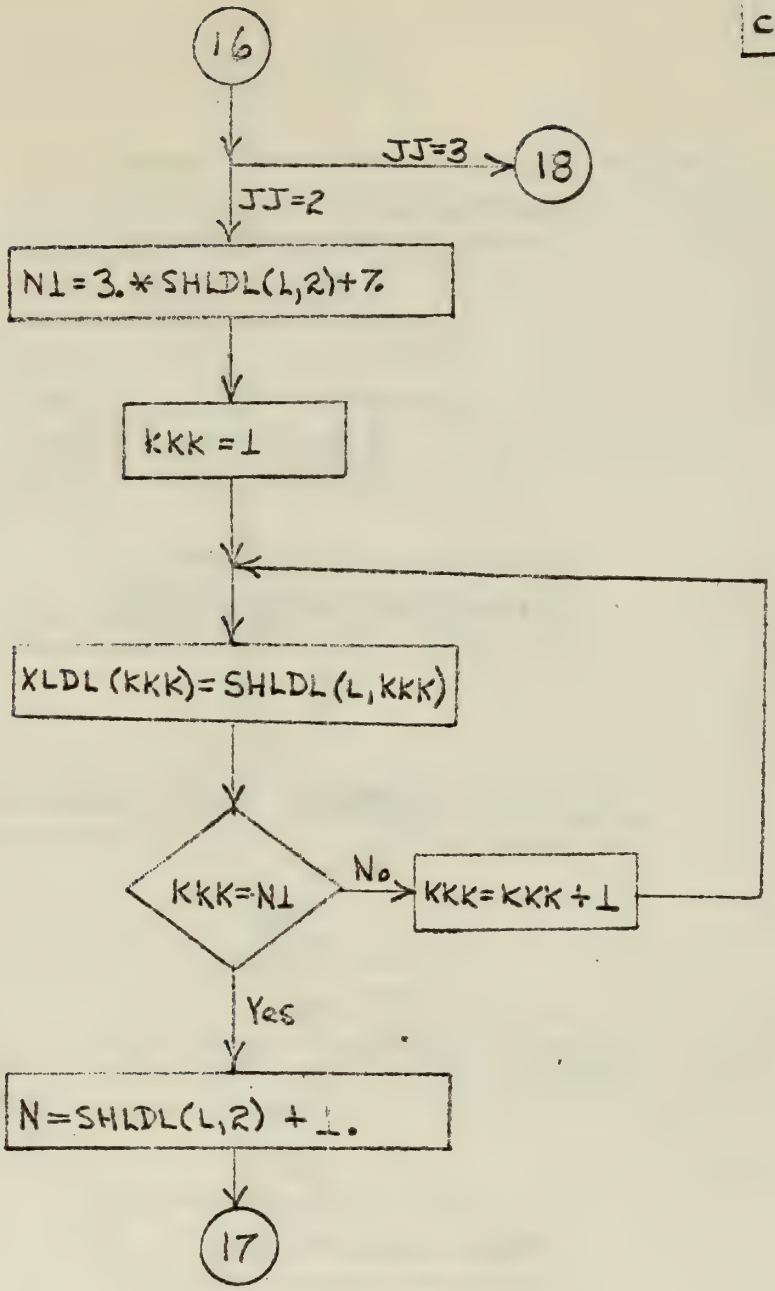

4. CONT'D
13
TD100 = SUCL(J,17)/ (SUCL(J2)*2025.7/60.)
MR=0
BNE=SUCL(J,4)
ANE=SUCL(J,5)
GNE=SUCL(J,6)
JJ=3
14
GNE=SHCL(J,6)
JJ=2
IDAM=0
12
MM=2
TD(1)=0.
NTØTLG=VAL(I,4)
15

4. CONT'D

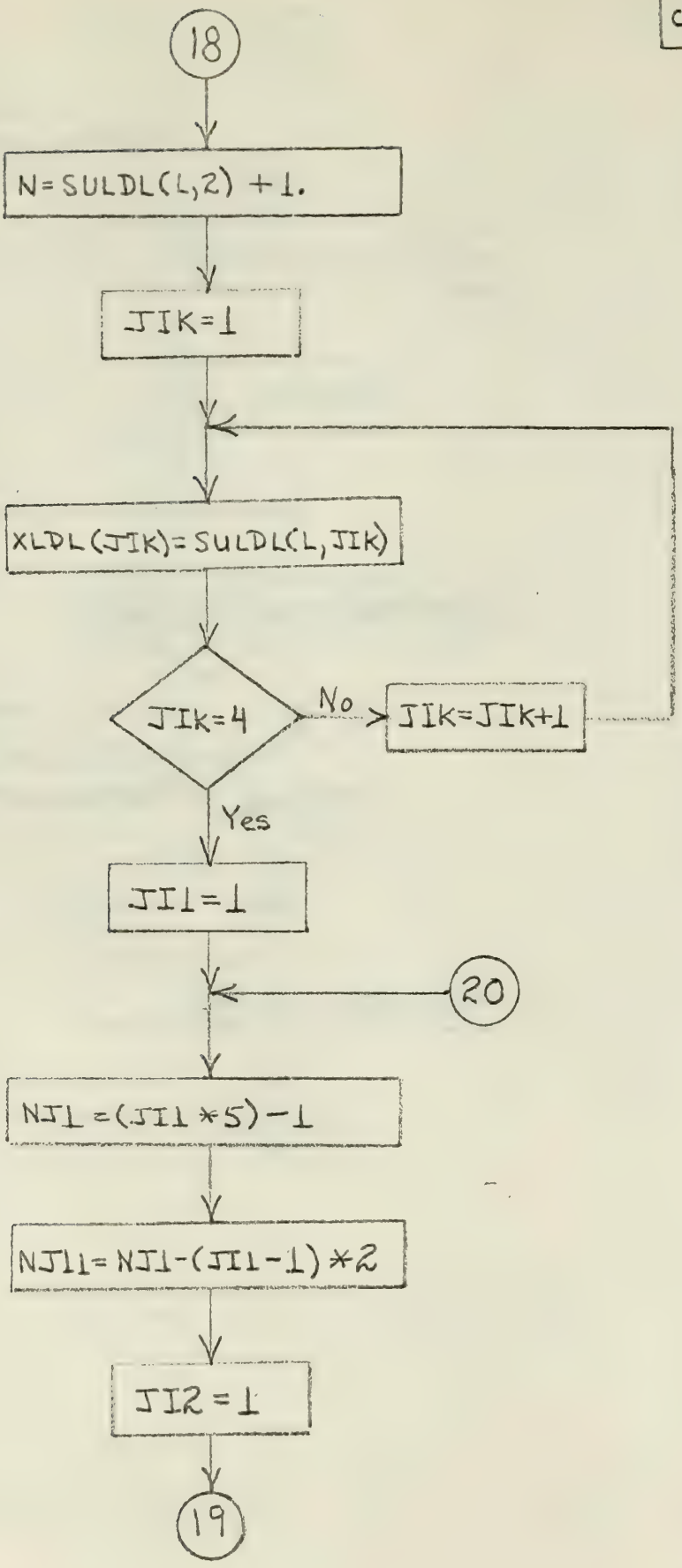



ENTRY
KK=1
24
CALL XTLEGN (XLCØD, L)
GØ TØ JJ
JJ=2,3
16
JJ=1.
N1= 3.*ACLDL(L,2)+7.
KKK=1
XLDL(KKK)=ACLDL(L,KKK)
KKK=N1
No
KKK=KKK+1
Yes
N=ACLDL(L,2)+1.
17

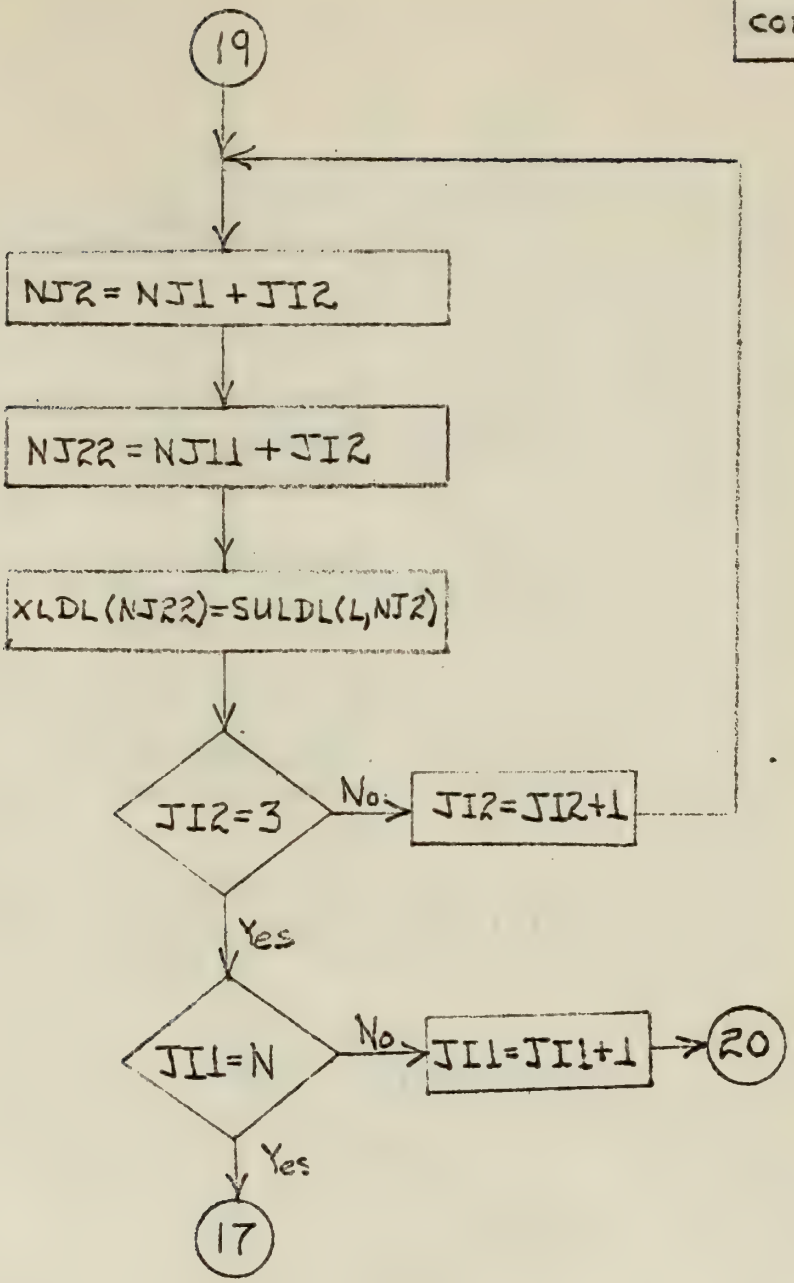
5. CON'TD


16
JJ=3
18
JJ=2
N1=3.*SHLDL(L,2)+7.
KKK=1
XLDL(KKK)=SHLDL(L,KKK)
KKK=N1
No
KKK=KKK+1
Yes
N=SHLDL(L,2)+1.
17

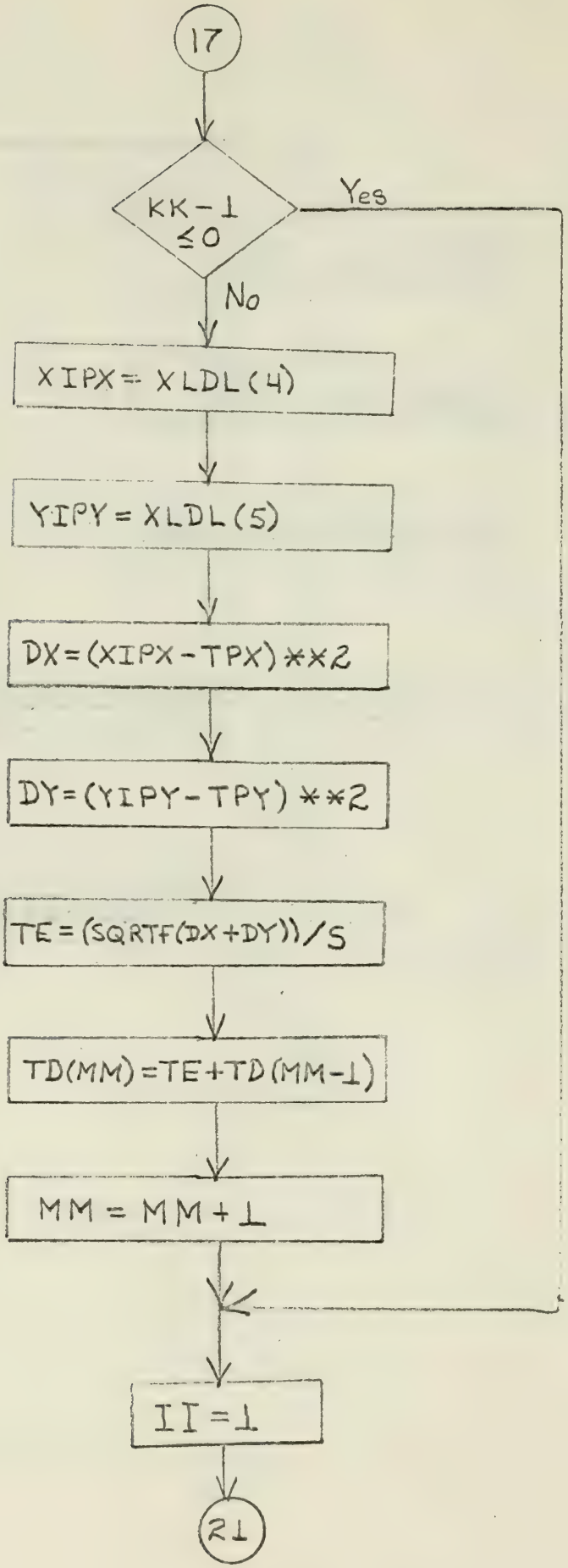
5. CONT'D


18
N=SULDL(L,2) +1.
JIK=1
XLDL(JIK)=SULDL(L,JIK)
JIk=4
No
JIk=JIK+1
Yes
JI1=1
20
NJ1=(JI1*5)-1
NJ11=NJ1-(JI1-1)*2
JI2=1
19

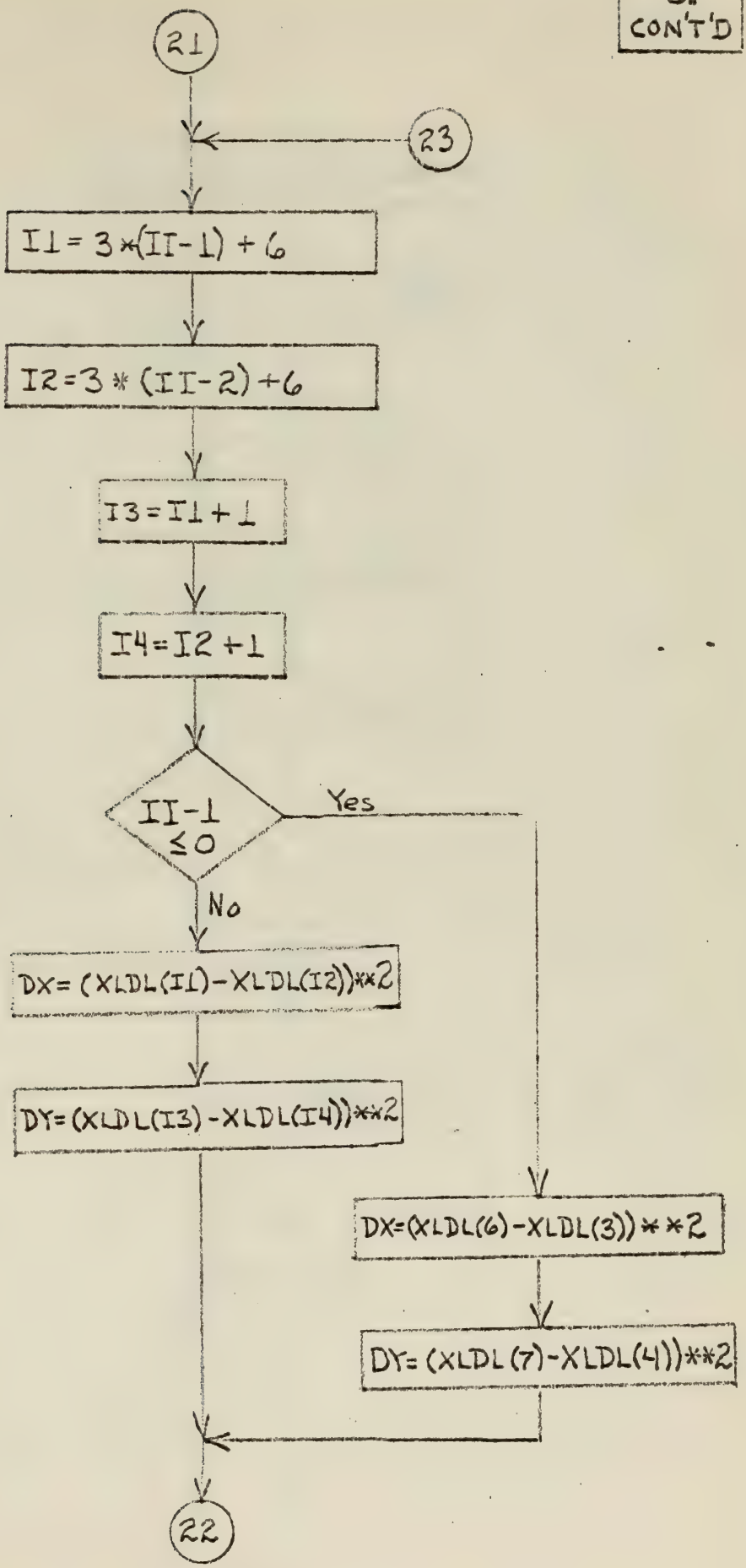
5. CON'TD


19
NJ2 = NJ1 + JI2
NJ22 = NJ11 + JI2
XLDL(NJ22)=SULDL(L,NJ2)
JI2=3
No
JI2=JI2+1
Yes
JI1=N
No
JI1=JI1+1
20
Yes
17

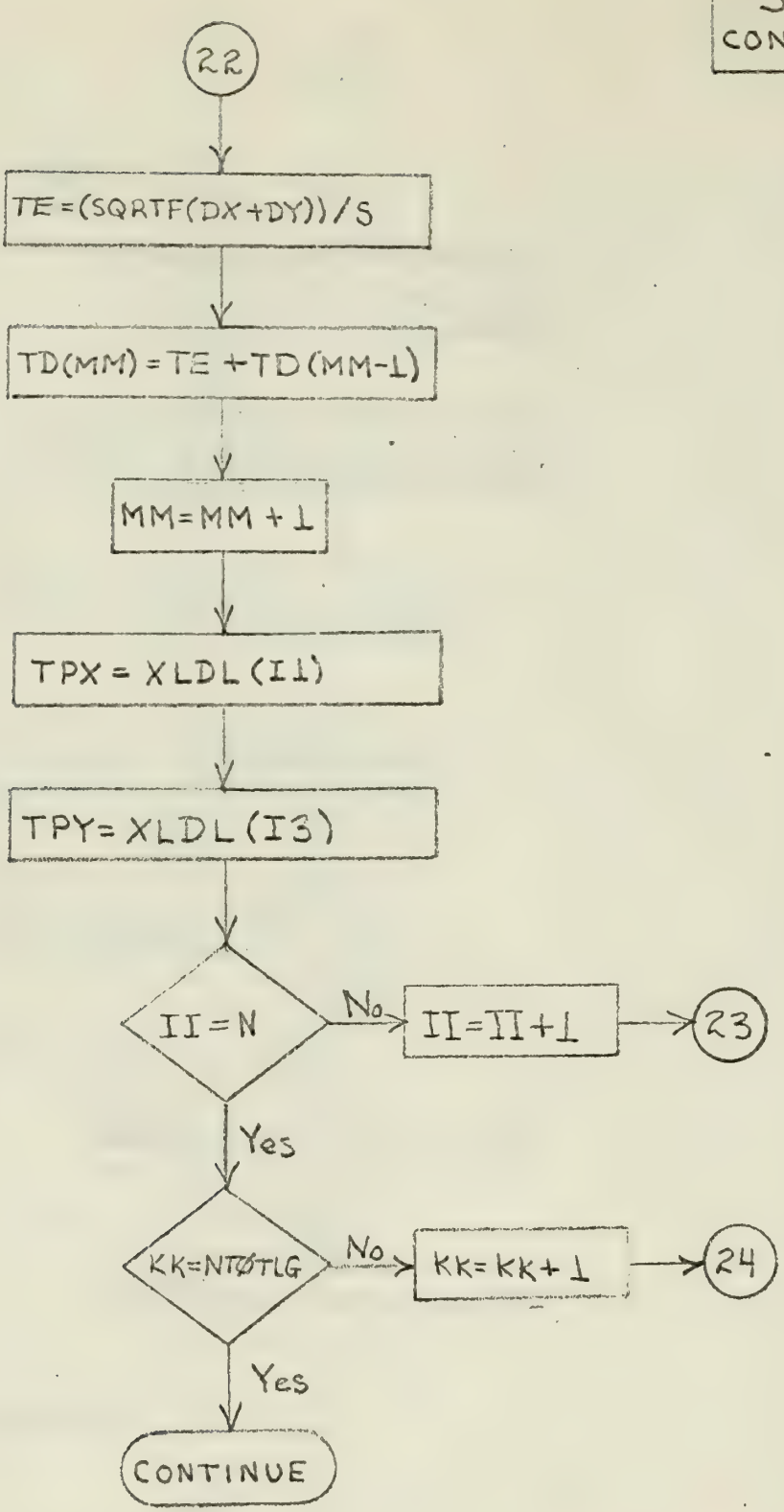
5.
CON'T'D

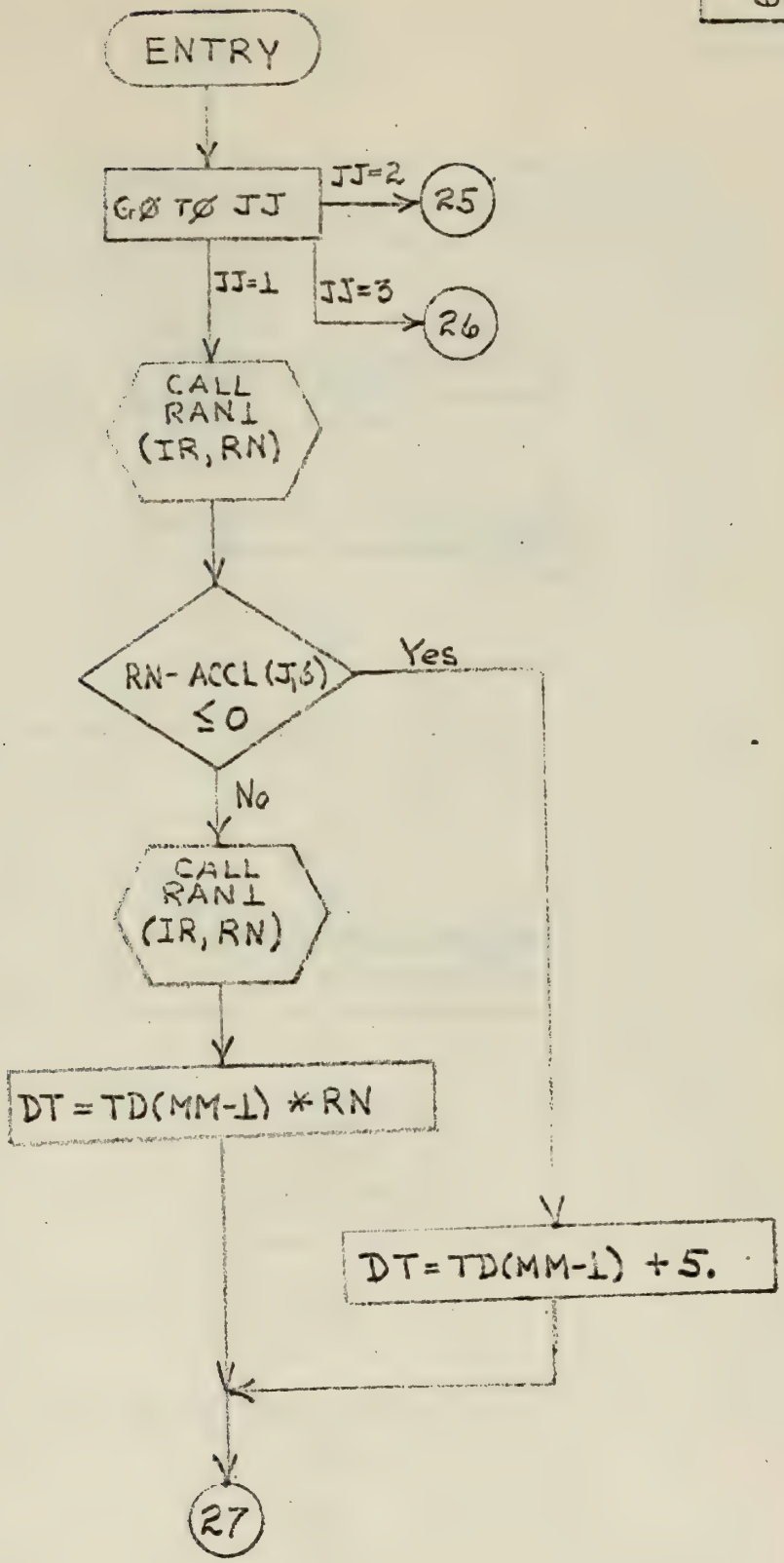

17
KK-1 ≤0
Yes
No
XIPX = XLDL(4)
YIPY = XLDL(5)
DX=(XIPX-TPX)**2
DY=(YIPY-TPY)**2
TE=(SQRTF(DX+DY))/S
TD(MM)=TE+TD(MM-1)
MM = MM+1
II=1
21

5. CON'T'D

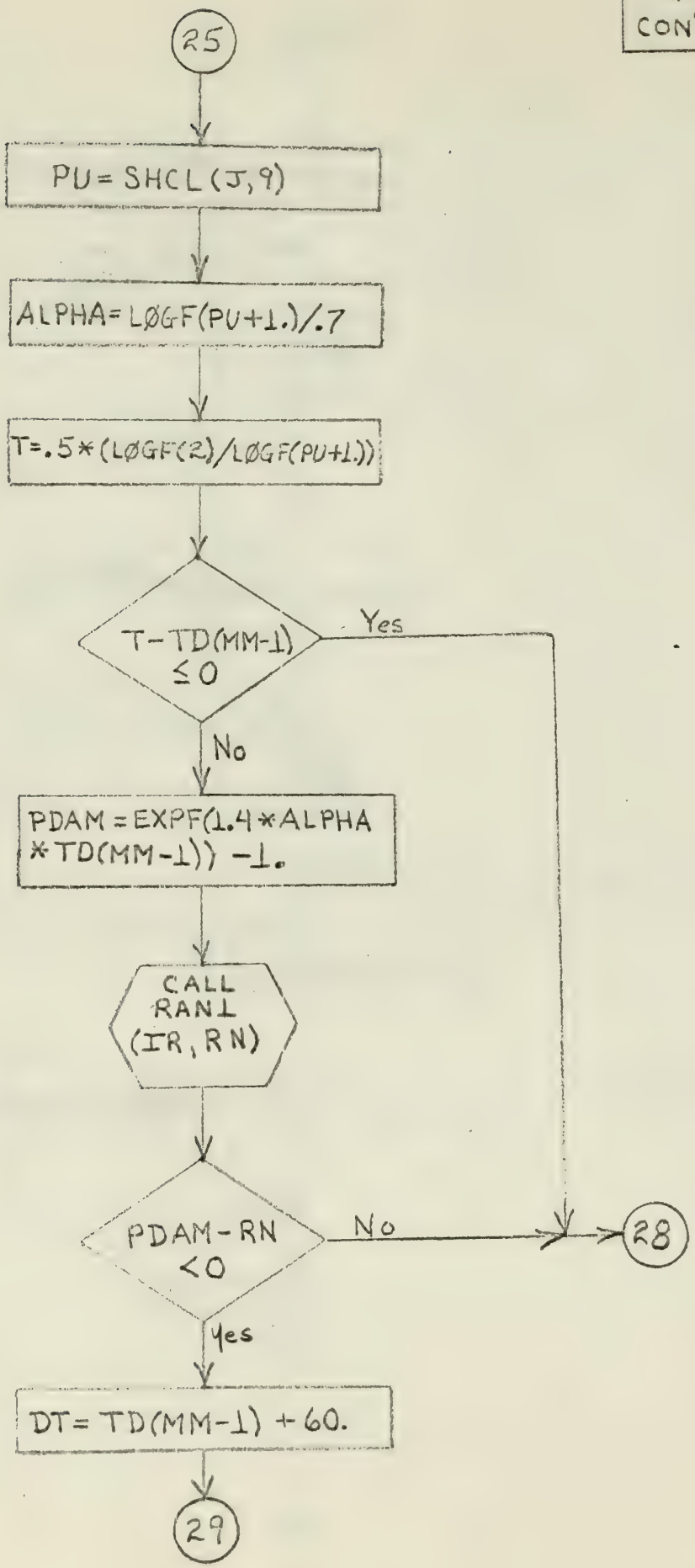

21
23
I1 = 3*(II-1) + 6
I2 = 3 * (II-2) + 6
I3 = I1 + 1
I4 = I2 + 1
II-1 ≤ 0
Yes
No
DX = (XLDL(I1) - XLDL(I2))**2
DY = (XLDL(I3) - XLDL(I4))**2
DX = (XLDL(6) - XLDL(3)) * * 2
DY = (XLDL(7) - XLDL(4))**2
22

5. CONT'D

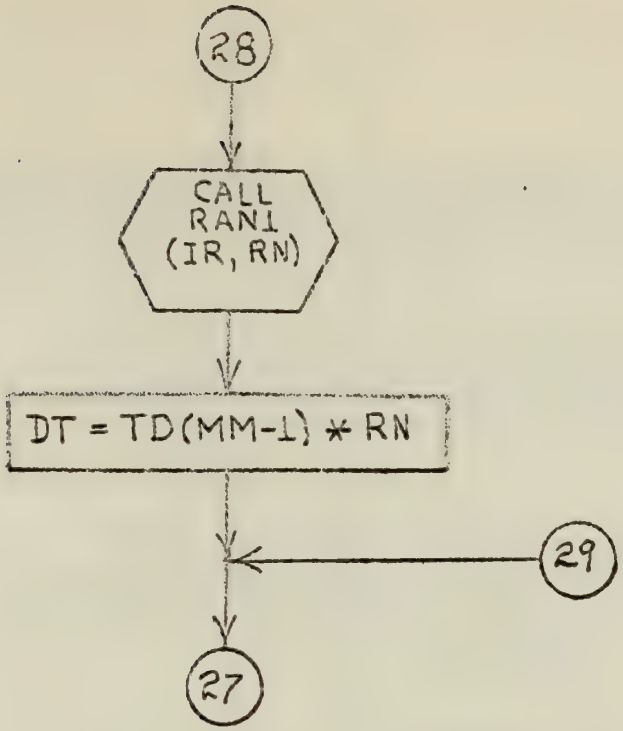

22
TE=(SQRTF(DX+DY))/S
TD(MM)=TE+TD(MM-1)
MM=MM+1
TPX=XLDL(I1)
TPY=XLDL(I3)
II=N
No
II=II+1
23
Yes
KK=NTØTLG
No
KK=KK+1
24
Yes
CONTINUE


ENTRY
GØ TØ JJ
JJ=2
25
JJ=1
JJ=3
26
CALL RANI (IR, RN)
RN - ACCL(J,6) ≤ 0
Yes
No
CALL RANI (IR, RN)
DT = TD(MM-1) * RN
DT = TD(MM-1) + 5.
27

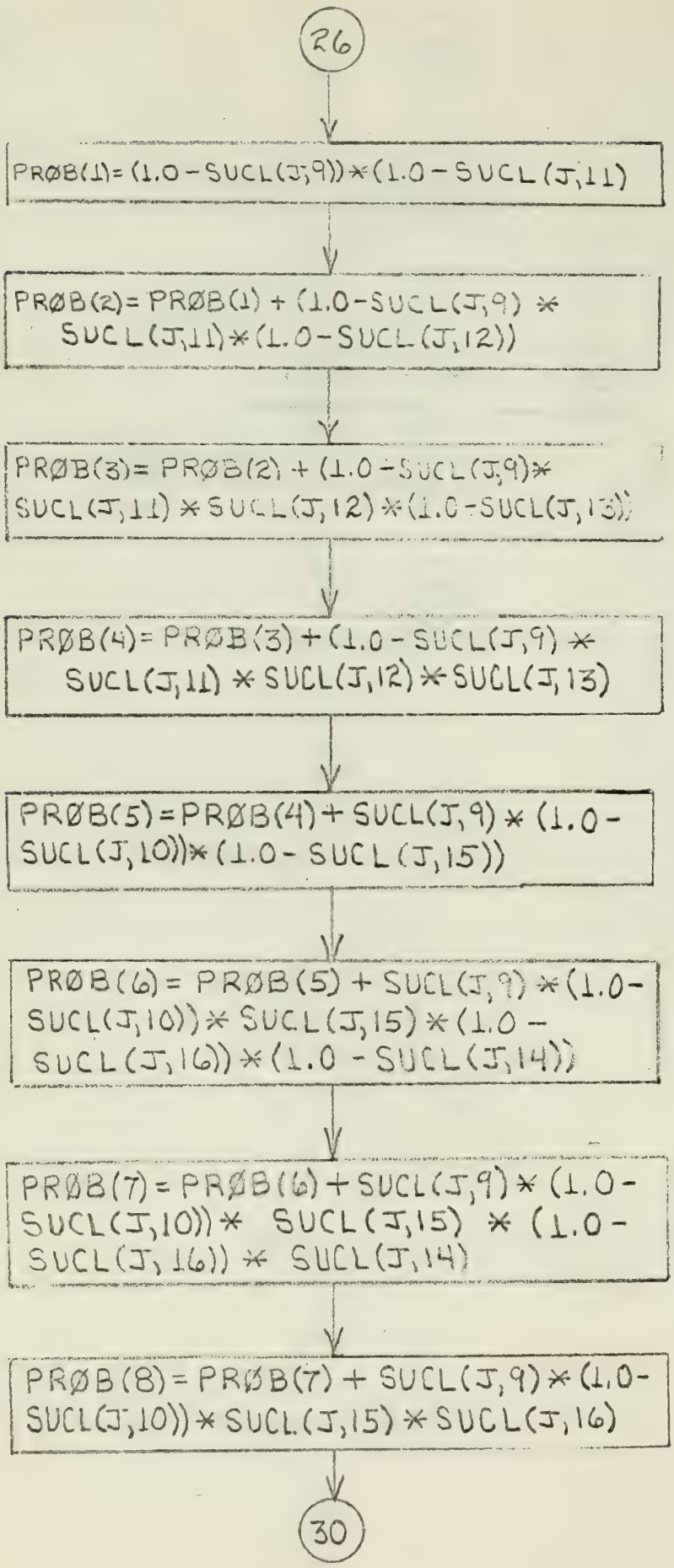
6. CON'T'D


25
PU = SHCL(J,9)
ALPHA = LØGF(PU+1.)/.7
T=.5*(LØGF(2.)/LØGF(PU+1.))
T-TD(MM-1) ≤0
Yes
No
PDAM = EXPF(1.4*ALPHA *TD(MM-1)) -1.
CALL RANL (IR, RN)
PDAM-RN <0
No
28
Yes
DT = TD(MM-1) + 60.
29

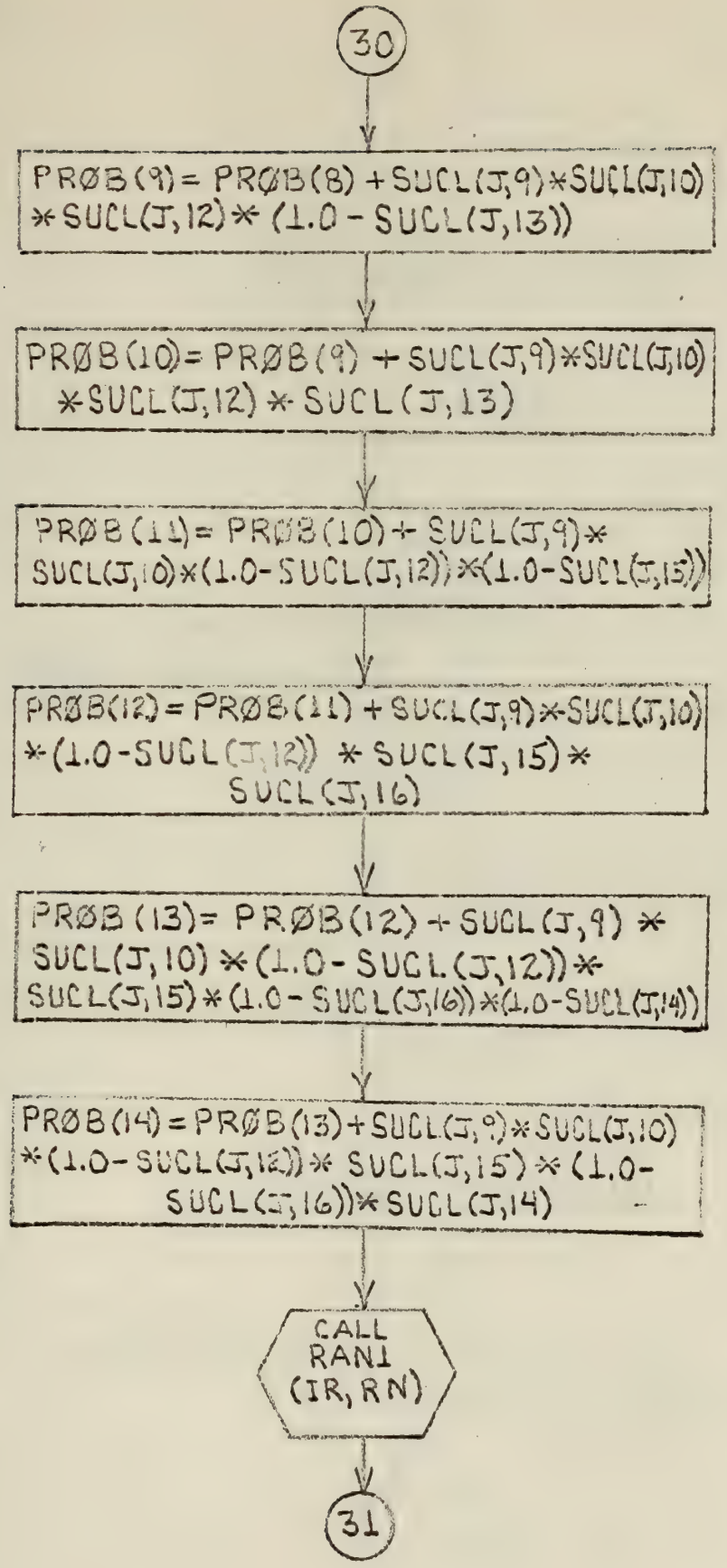
6. CON'T'D

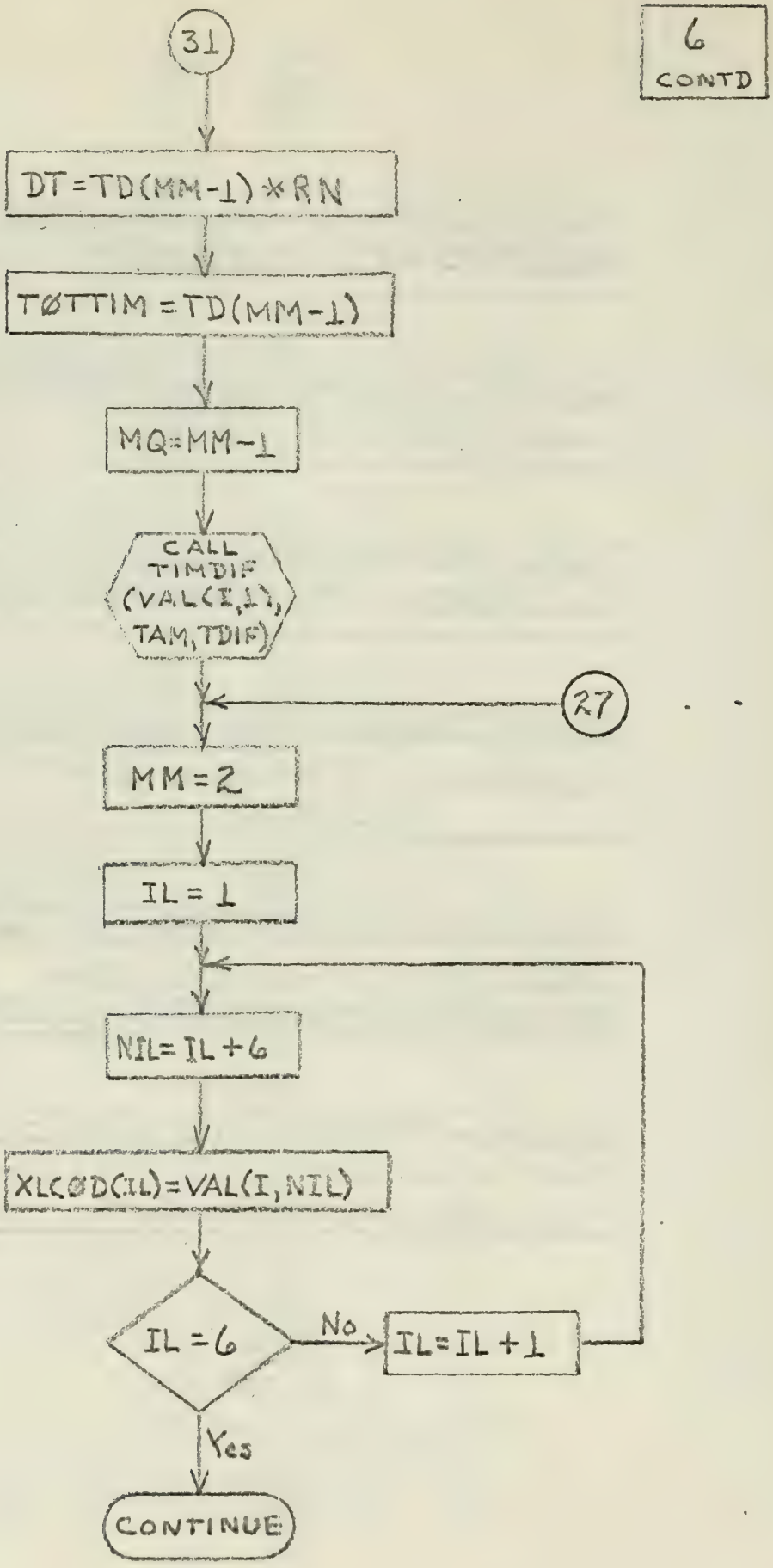


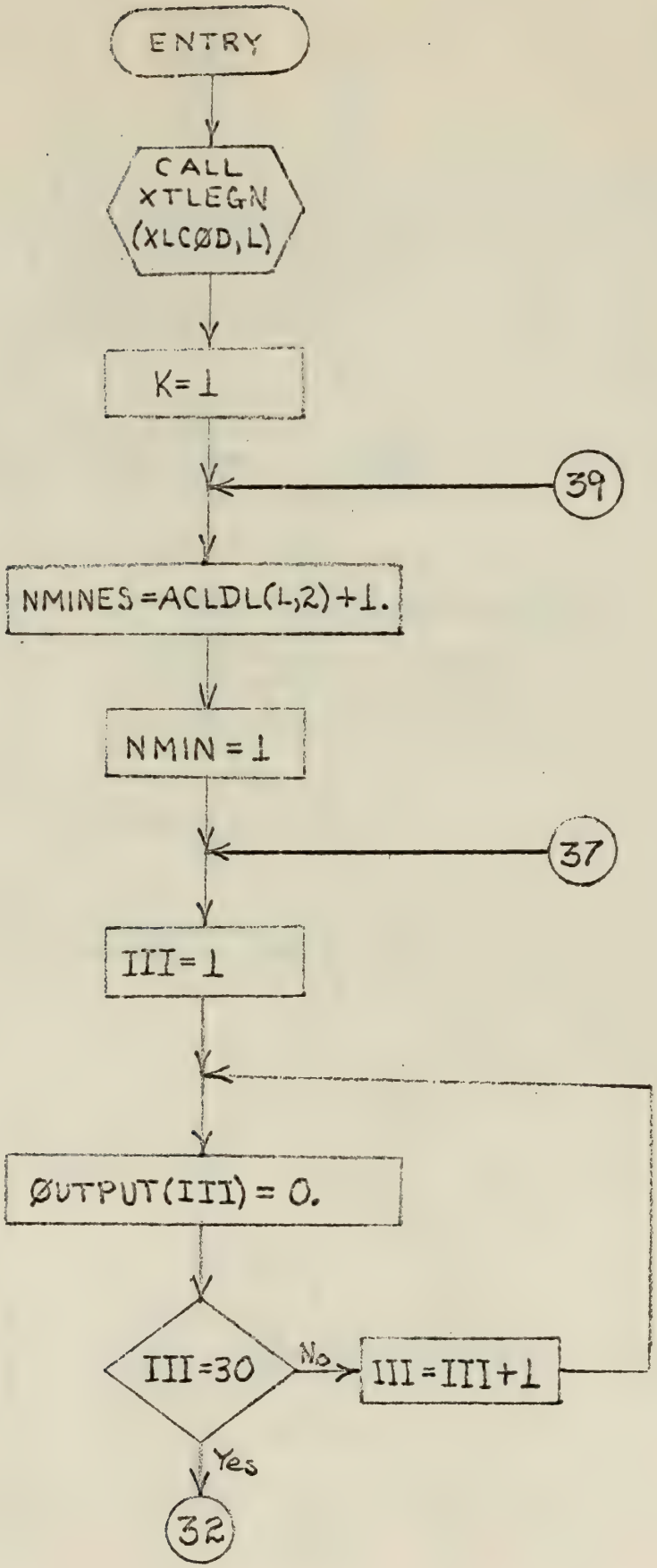
6 CONT'D

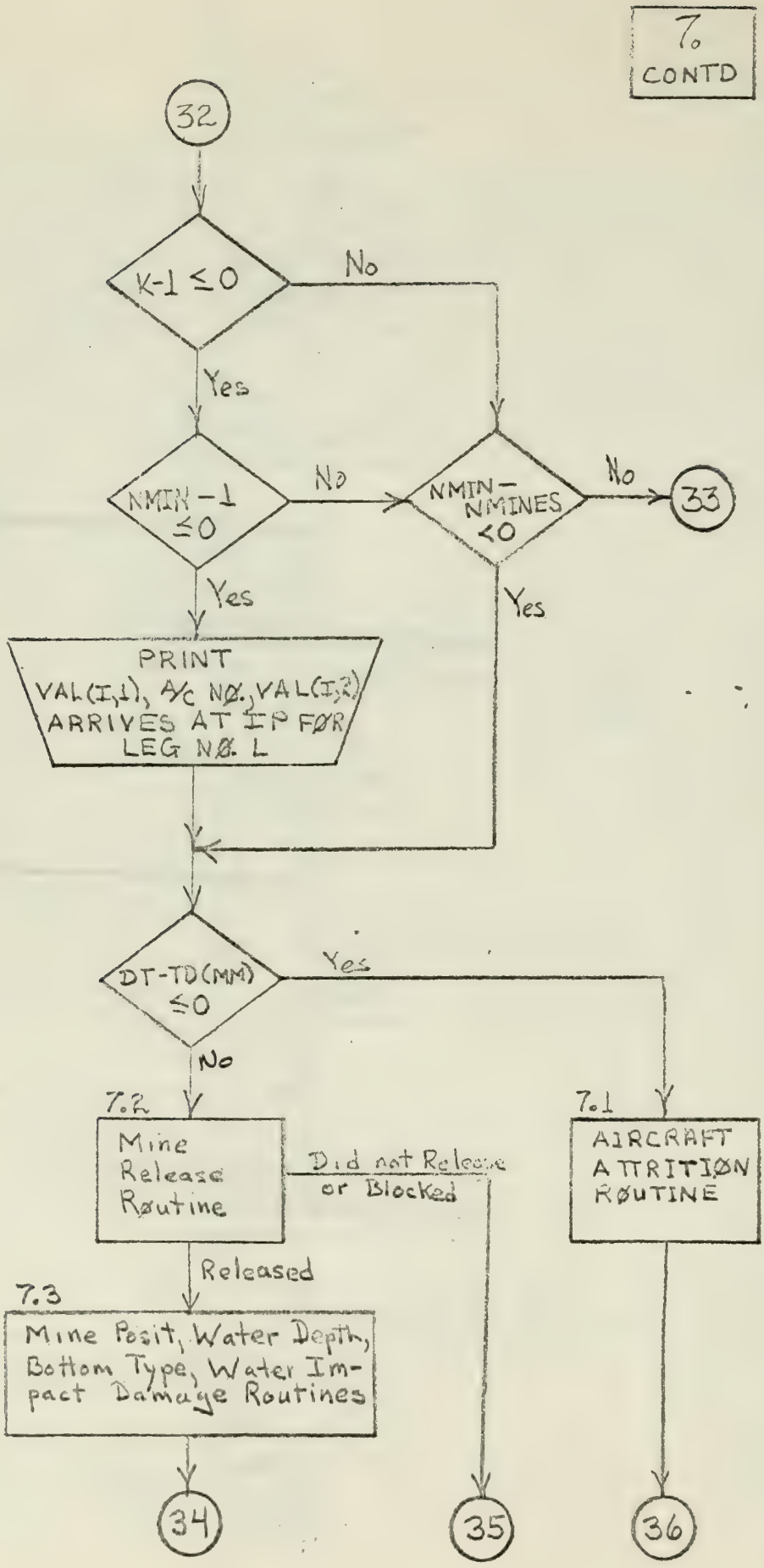

26
PRØB(1)=(1.0-SUCL(J,9))*(1.0-SUCL(J,11)
PRØB(2)=PRØB(1)+(1.0-SUCL(J,9)*SUCL(J,11)*(1.0-SUCL(J,12))
PRØB(3)=PRØB(2)+(1.0-SUCL(J,9)*SUCL(J,11)*SUCL(J,12)*(1.0-SUCL(J,13))
PRØB(4)=PRØB(3)+(1.0-SUCL(J,9)*SUCL(J,11)*SUCL(J,12)*SUCL(J,13)
PRØB(5)=PRØB(4)+SUCL(J,9)*(1.0-SUCL(J,10))*(1.0-SUCL(J,15))
PRØB(6)=PRØB(5)+SUCL(J,9)*(1.0-SUCL(J,10))*SUCL(J,15)*(1.0-SUCL(J,16))*(1.0-SUCL(J,14))
PRØB(7)=PRØB(6)+SUCL(J,9)*(1.0-SUCL(J,10))*SUCL(J,15)*(1.0-SUCL(J,16))*SUCL(J,14)
PRØB(8)=PRØB(7)+SUCL(J,9)*(1.0-SUCL(J,10))*SUCL(J,15)*SUCL(J,16)
30

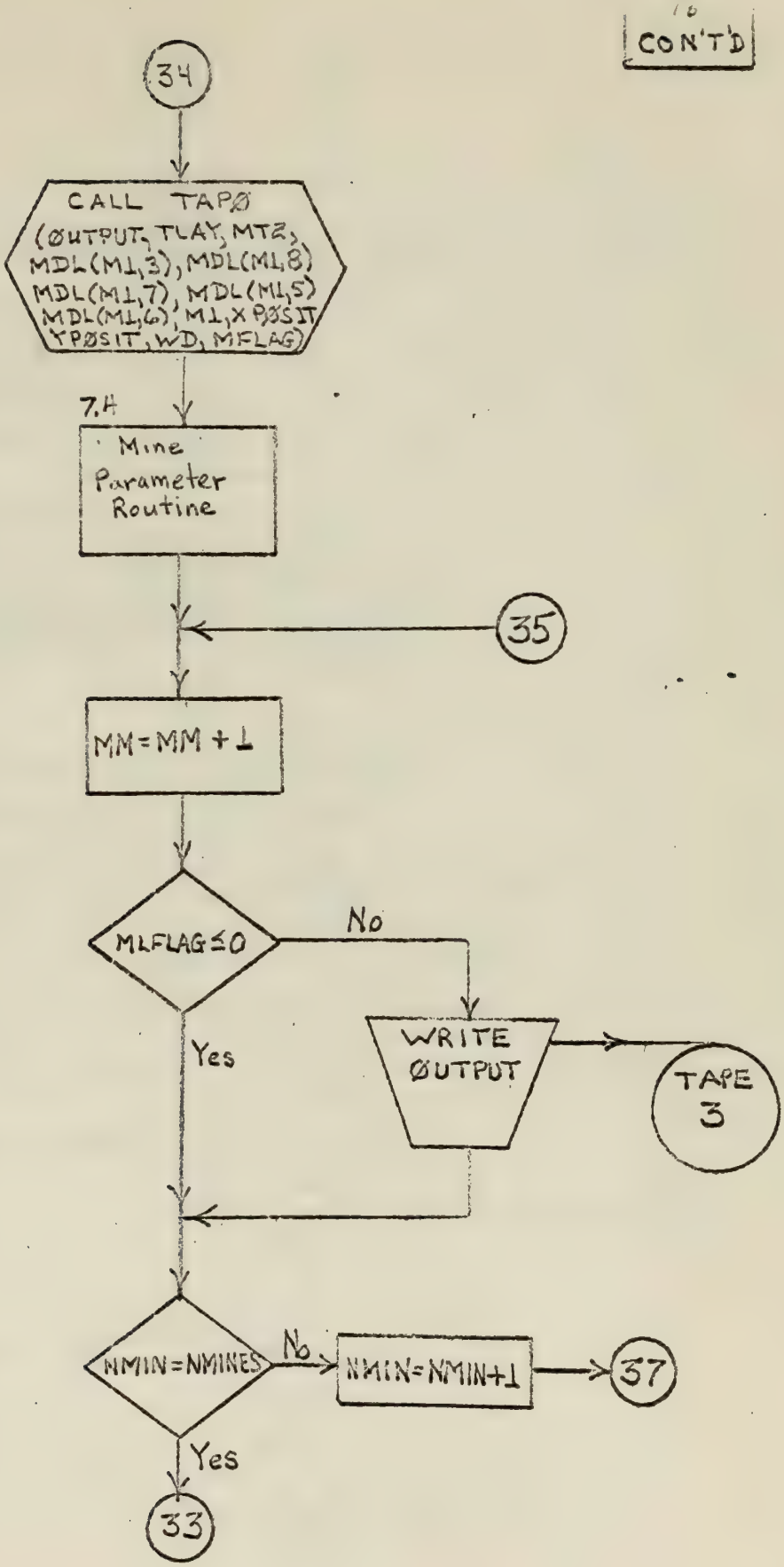
6. cont'd


30
PRØB(9) = PRØB(8) + SUCL(J,9)*SUCL(J,10) *SUCL(J,12)* (1.0 - SUCL(J,13))
PRØB(10) = PRØB(9) + SUCL(J,9)*SUCL(J,10) *SUCL(J,12) * SUCL(J,13)
PRØB(11) = PRØB(10) + SUCL(J,9)* SUCL(J,10)*(1.0-SUCL(J,12))*(1.0-SUCL(J,15))
PRØB(12) = PRØB(11) + SUCL(J,9)*SUCL(J,10) *(1.0-SUCL(J,12)) * SUCL(J,15) * SUCL(J,16)
PRØB(13) = PRØB(12) + SUCL(J,9) * SUCL(J,10) * (1.0 - SUCL(J,12)) * SUCL(J,15) * (1.0 - SUCL(J,16)) * (1.0 - SUCL(J,14))
PRØB(14) = PRØB(13) + SUCL(J,9) * SUCL(J,10) *(1.0 - SUCL(J,12)) * SUCL(J,15) * (1.0 - SUCL(J,16)) * SUCL(J,14)
CALL RANI (IR, RN)
31

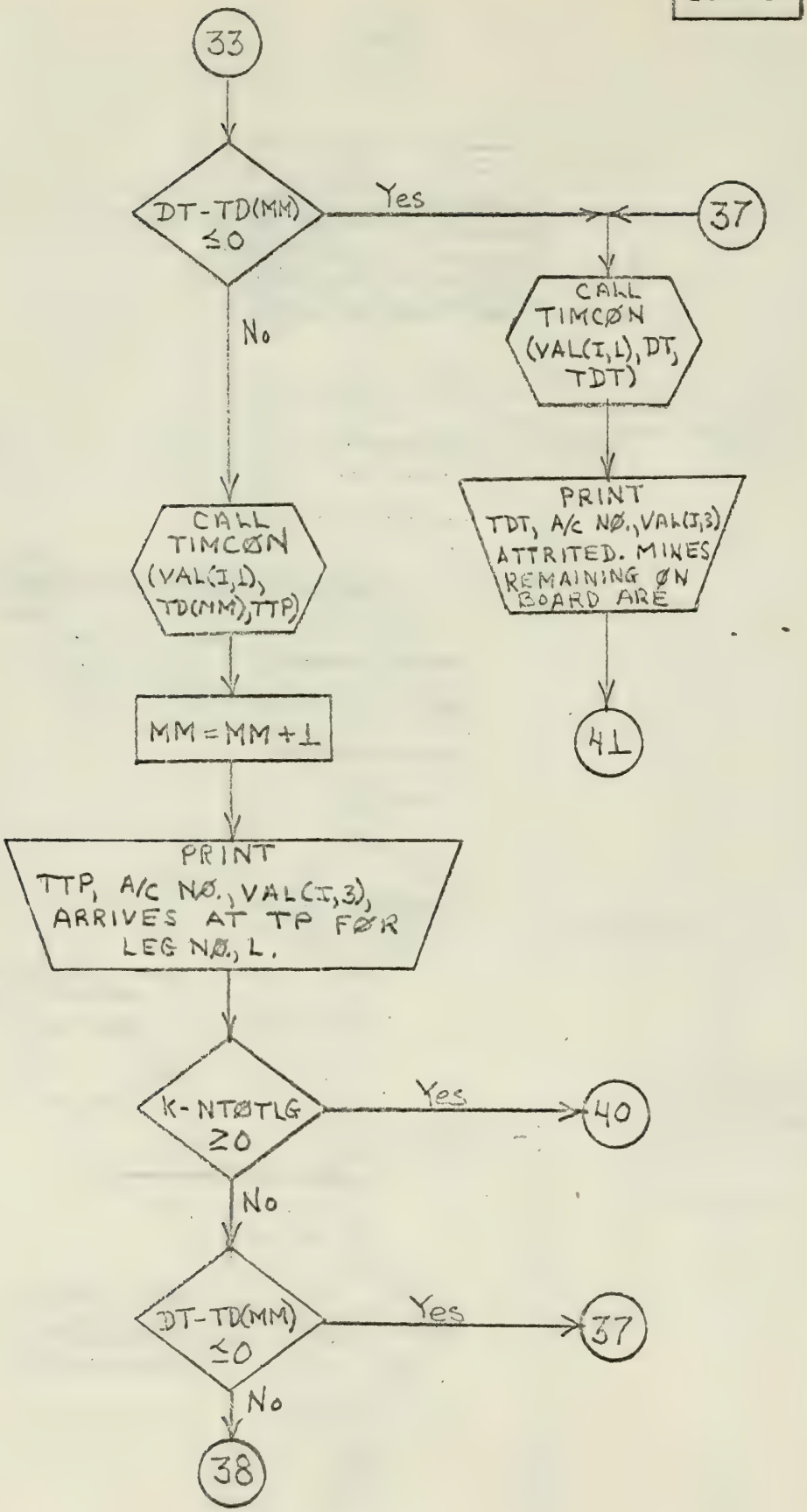

6
CONTD
31
DT=TD(MM-1)*RN
TØTTIM=TD(MM-1)
MQ=MM-1
CALL TIMDIF (VAL(I,1), TAM,TDIF)
27
MM=2
IL=1
NIL=IL+6
XLCØD(IL)=VAL(I,NIL)
IL=6
No
IL=IL+1
Yes
CONTINUE


ENTRY
CALL XTLEGN (XLCØD,L)
K=1
39
NMINES=ACLDL(L,2)+1.
NMIN=1
37
III=1
ØUTPUT(III)=0.
III=30
No
III=III+1
Yes
32

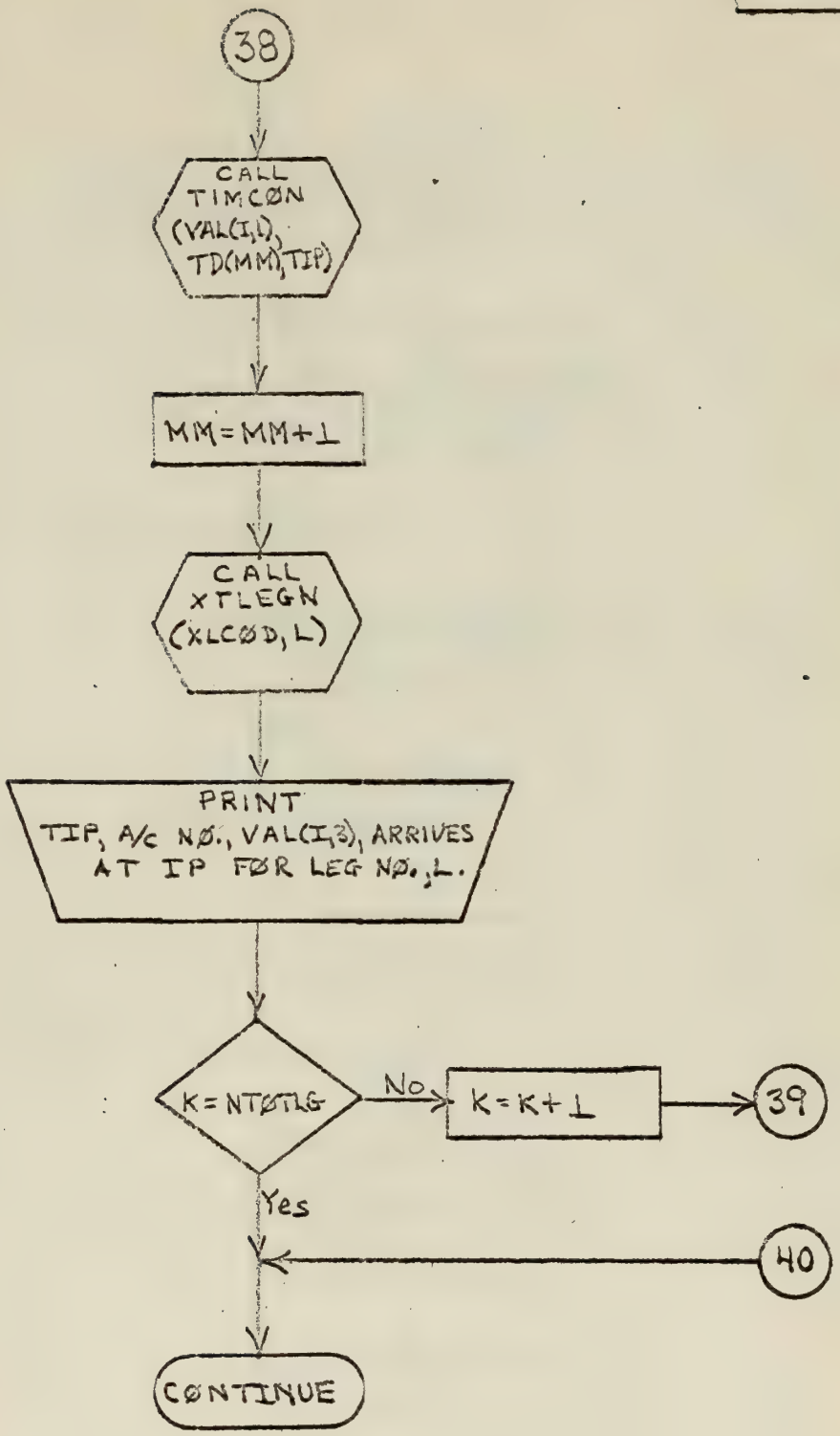

7.
CONTD
32
K-1 ≤0
No
Yes
NMIN -1 ≤0
No
NMIN - NMINES <0
No
33
Yes
Yes
PRINT VAL(I,1), A/C NØ., VAL(I,2) ARRIVES AT IP FØR LEG NØ. L
DT-TD(MM) ≤0
Yes
No
7.2
Mine Release Routine
Did not Release or Blocked
7.1
AIRCRAFT ATTRITIØN RØUTINE
Released
7.3
Mine Posit, Water Depth, Bottom Type, Water Impact Damage Routines
34
35
36


CON'T'D
34
CALL TAPØ
(ØUTPUT, TLAY, MTZ,
MDL(ML,3), MDL(ML,8)
MDL(ML,7), MDL(ML,5)
MDL(ML,6), ML, XPØSIT
YPØSIT, WD, MFLAG)
7.4
Mine
Parameter
Routine
35
MM=MM+1
MLFLAG≤0
No
Yes
WRITE
ØUTPUT
TAPE
3
NMIN=NMINES
No
NMIN=NMIN+1
37
Yes
33

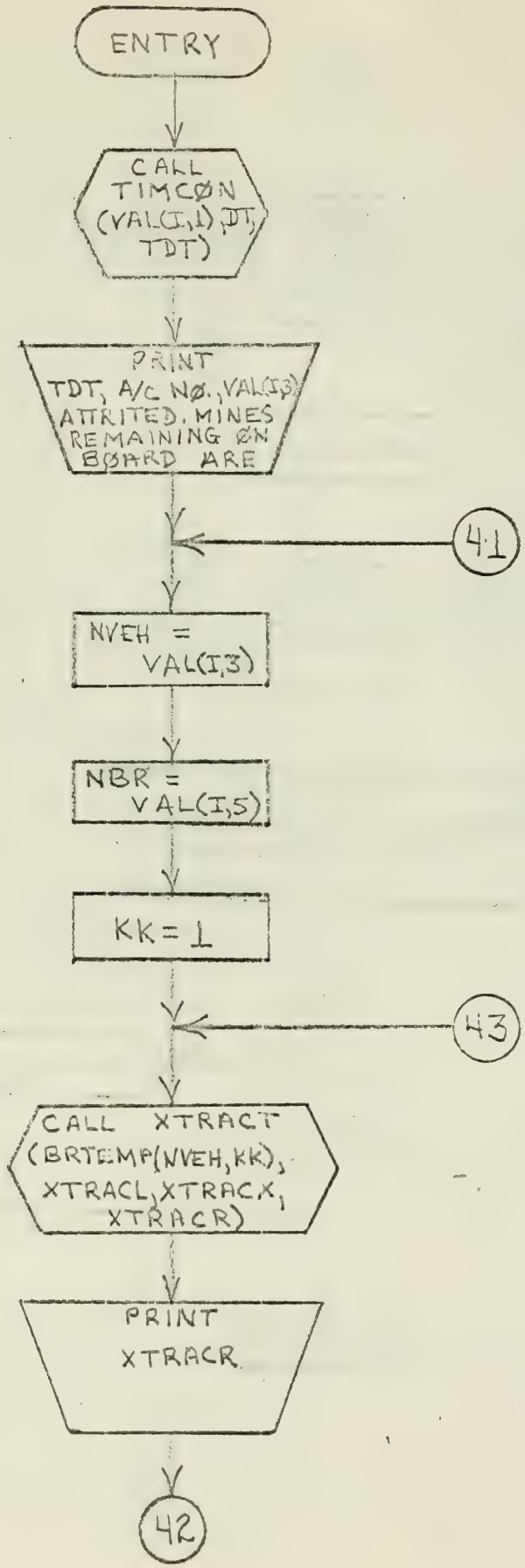
7. CON'T'D


33
DT-TD(MM) ≤0
Yes
37
No
CALL TIMCØN (VAL(I,L), DT, TDT)
PRINT TDT, A/C NØ., VAL(I,3) ATTRITED. MINES REMAINING ØN BOARD ARE
41
CALL TIMCØN (VAL(I,1), TD(MM), TTP)
MM = MM + 1
PRINT TTP, A/C NØ., VAL(I,3), ARRIVES AT TP FØR LEG NØ., L.
K - NTØTLG ≥0
Yes
40
No
DT-TD(MM) ≤0
Yes
37
No
38

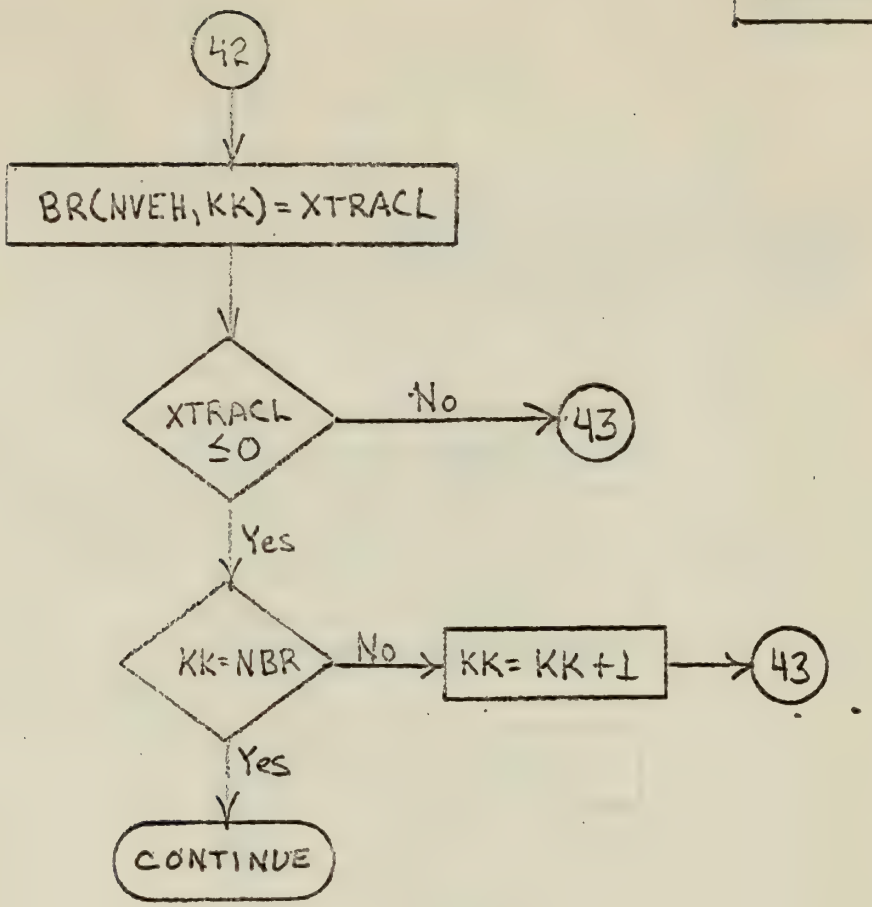
7.
CONT'D

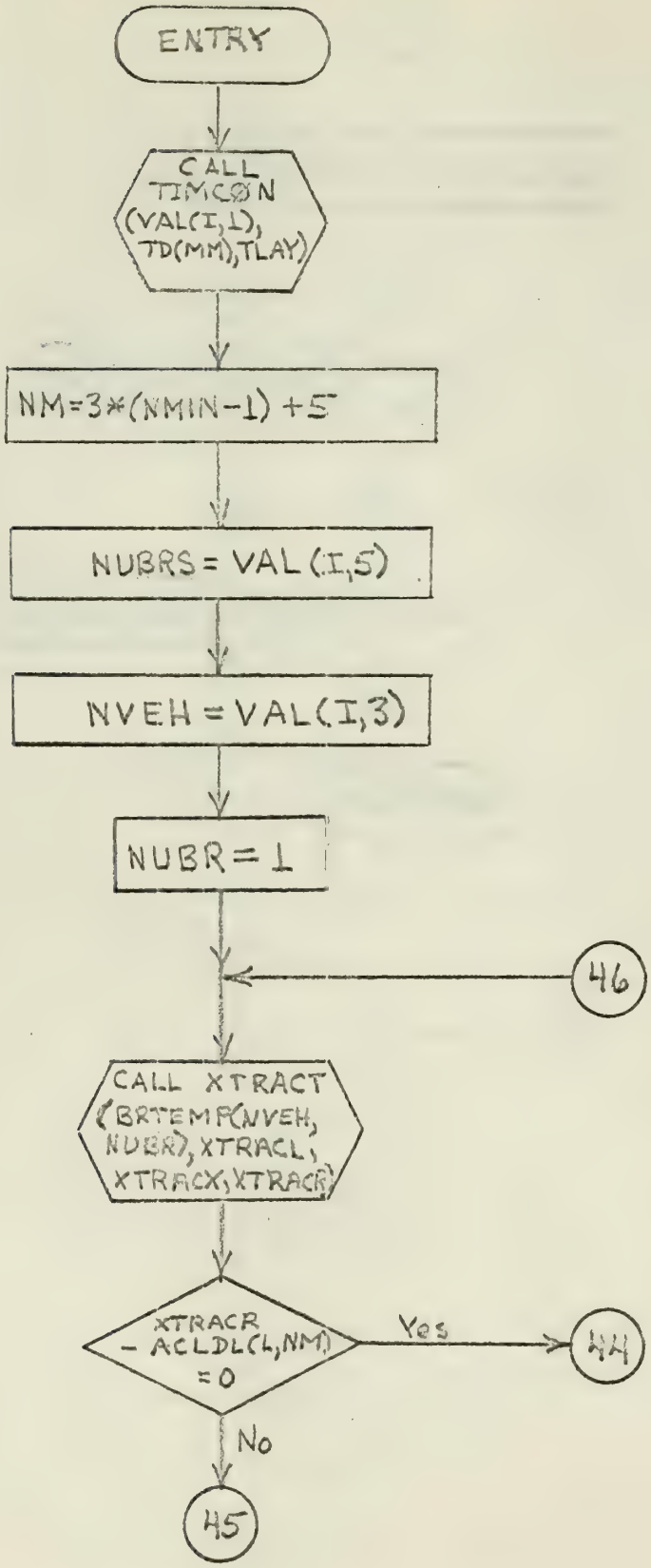

38
CALL TIMCØN (VAL(I,1), TD(MM), TIP)
MM=MM+1
CALL XTLEGN (XLCØD, L)
PRINT TIP, A/C NØ., VAL(I,3), ARRIVES AT IP FØR LEG NØ., L.
K=NTØTLG
No
K=K+1
39
Yes
40
CØNTINUE

7.1

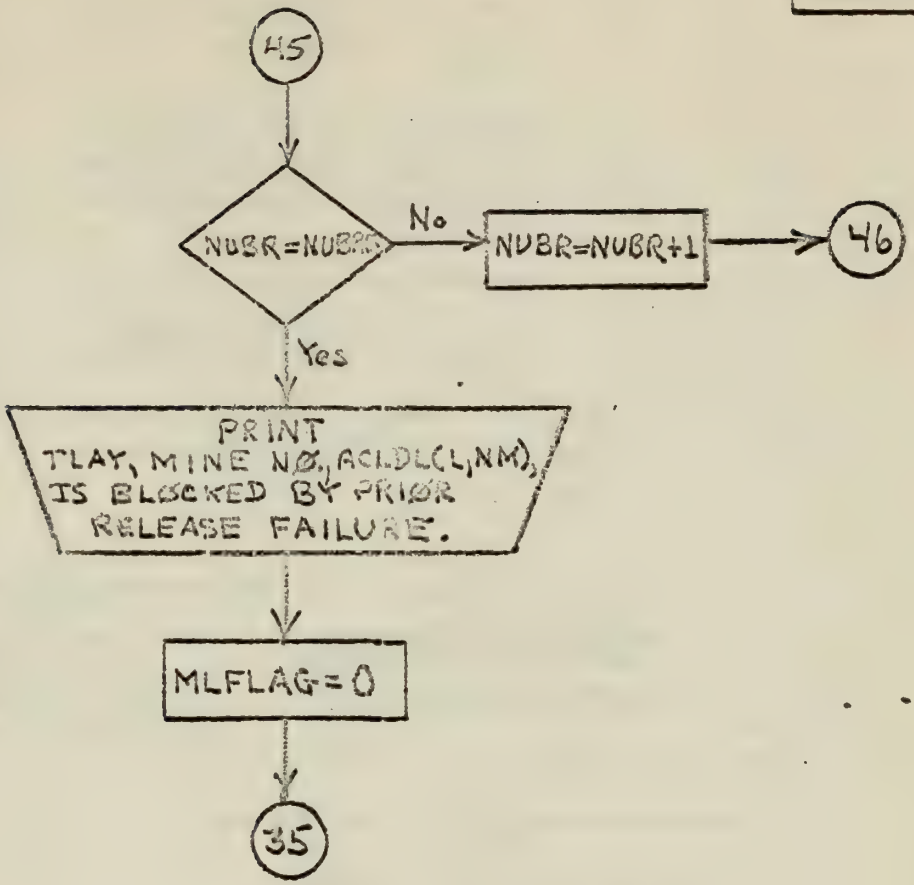

ENTRY
CALL TIMCØN (VAL(I,1), DT, TDT)
PRINT TDT, A/C NØ., VAL(I3) ATTRITED. MINES REMAINING ØN BØARD ARE
41
NVEH = VAL(I,3)
NBR = VAL(I,5)
KK = 1
43
CALL XTRACT (BRTEMP(NVEH,KK), XTRACL, XTRACX, XTRACR)
PRINT XTRACR
42

7.1
CONT'D

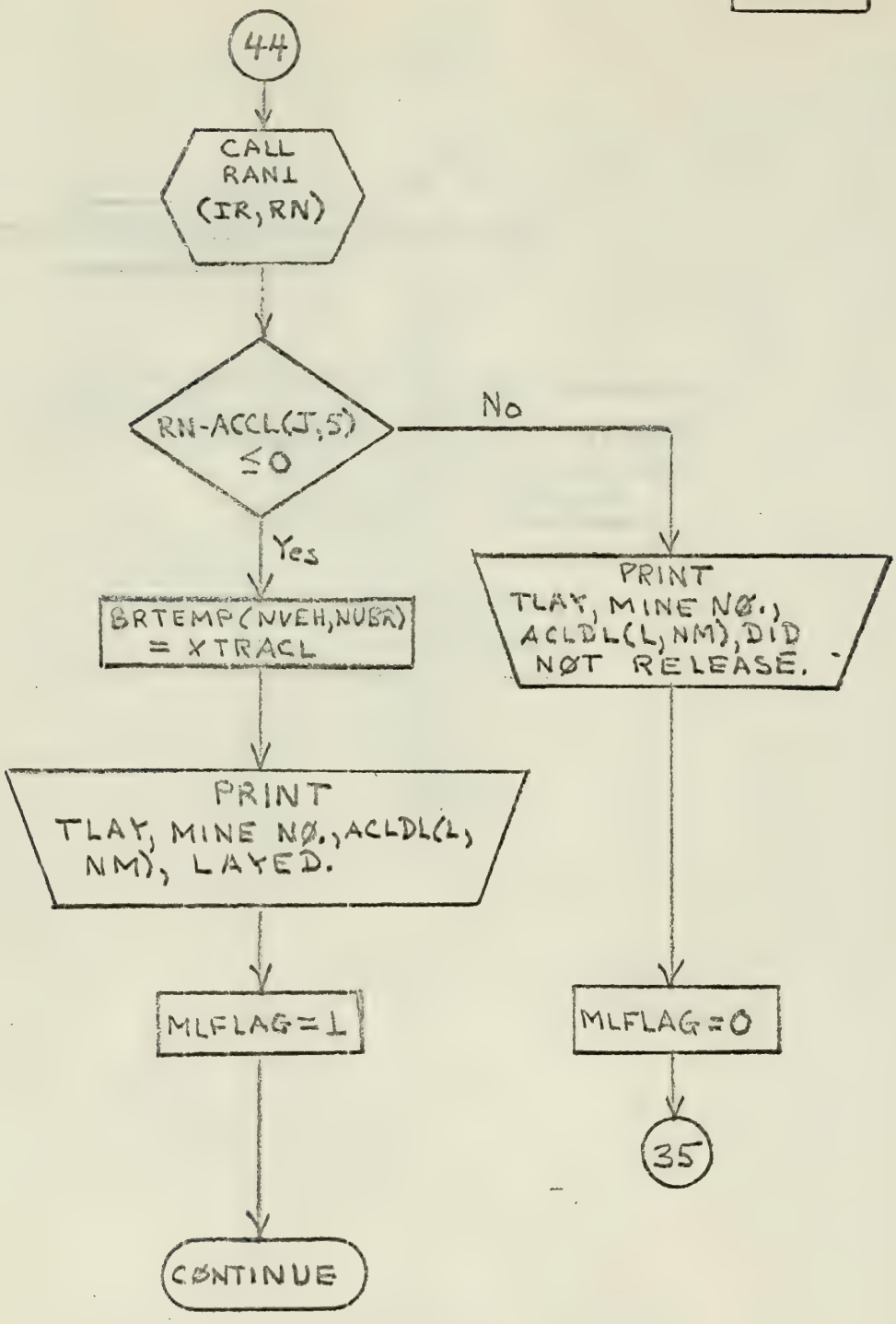

42
BR(NVEH, KK) = XTRACL
XTRACL ≤ 0
No
43
Yes
KK = NBR
No
KK = KK + 1
43
Yes
CONTINUE

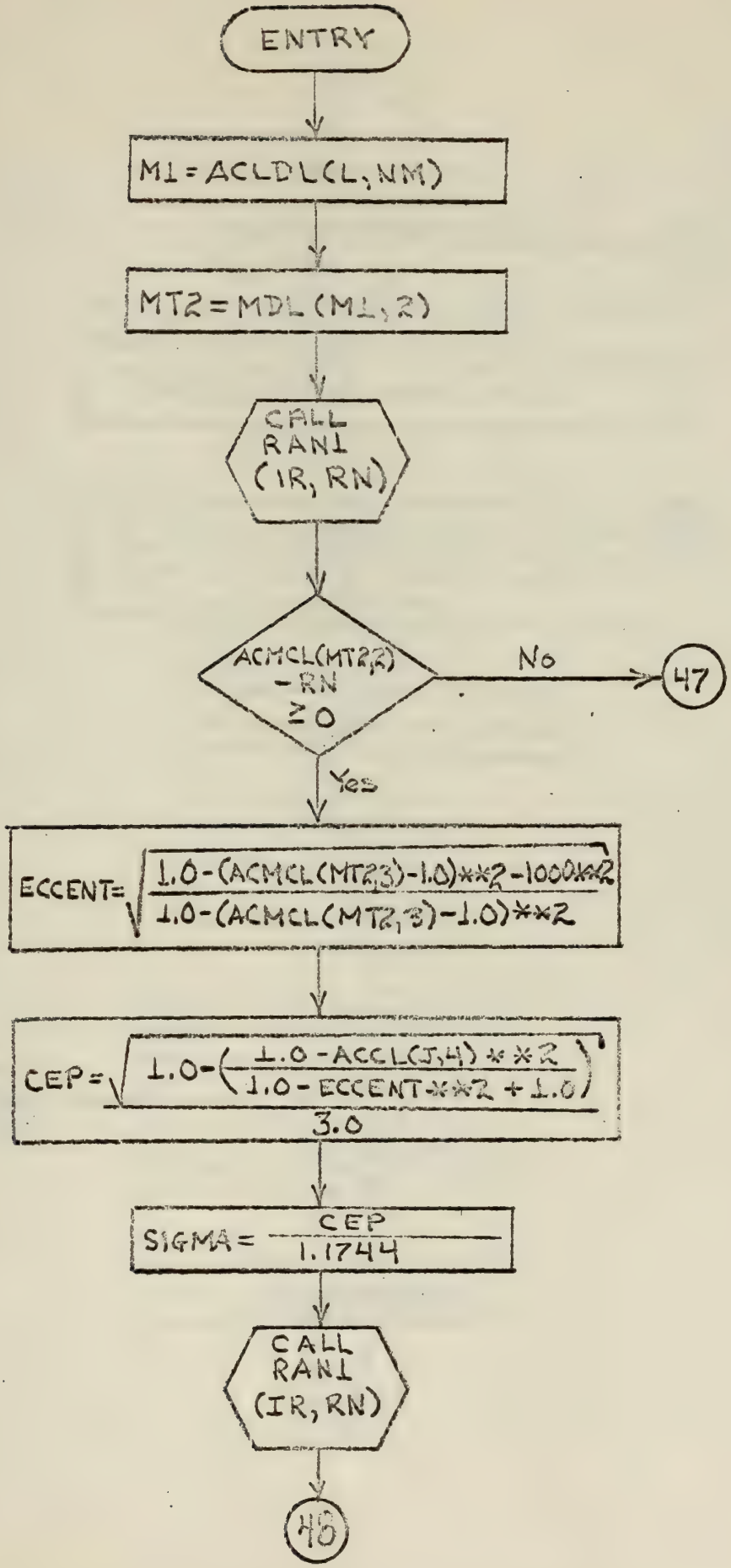

ENTRY
CALL TIMCØN (VAL(I,L), TD(MM), TLAY)
NM=3*(NMIN-1)+5
NUBRS = VAL(I,5)
NVEH = VAL(I,3)
NUBR = 1
46
CALL XTRACT (BRTEMP(NVEH, NUBR), XTRACL, XTRACX, XTRACR)
XTRACR - ACLDL(L,NM) = 0
Yes
44
No
45

7.2 CONT'D

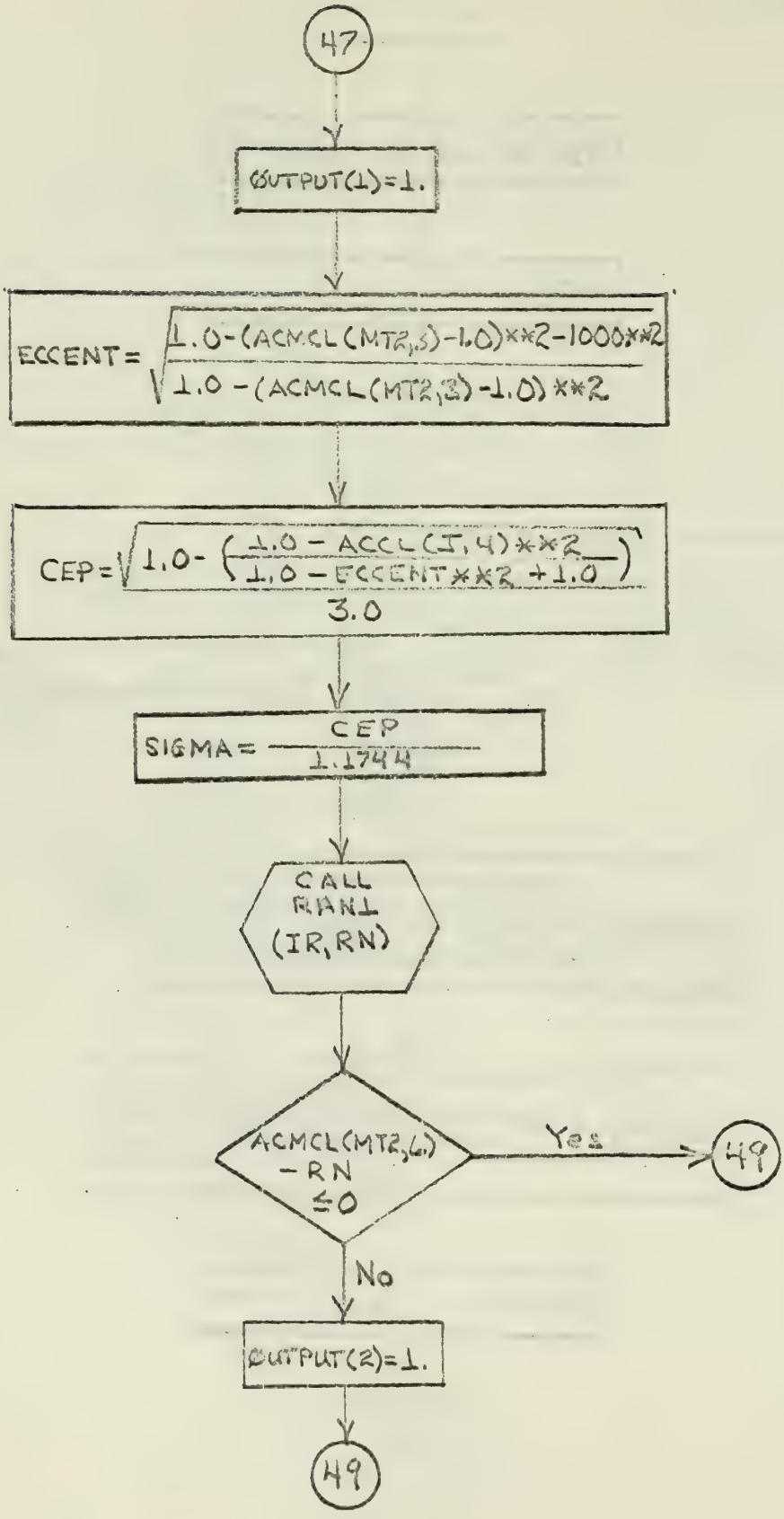

45
NUBR=NUBRS
No
NUBR=NUBR+1
46
Yes
PRINT
TLAY, MINE NØ., ACLDL(L,NM),
IS BLOCKED BY PRIØR
RELEASE FAILURE.
MLFLAG=0
35

7.2
CONTD

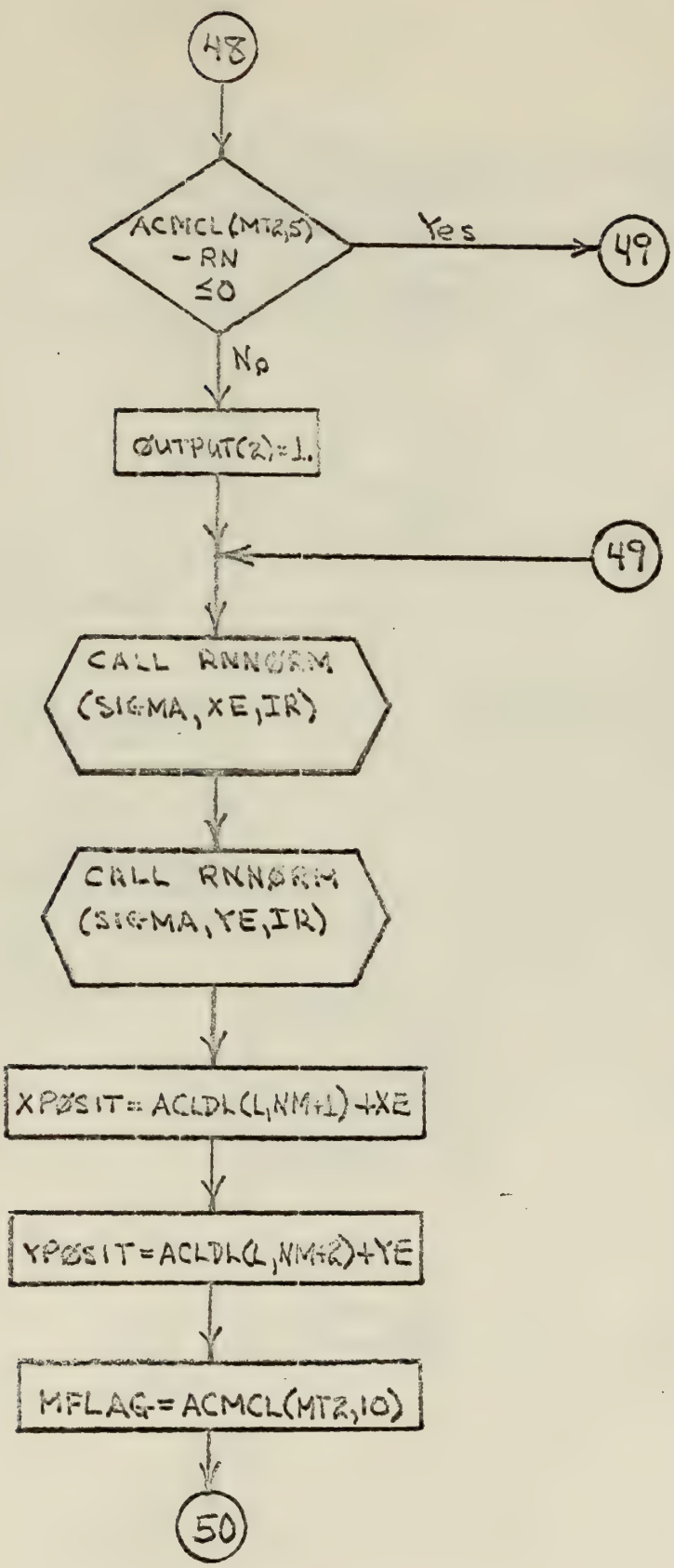

44
CALL RANI (IR, RN)
RN-ACCL(J,5) ≤0
No
Yes
BRTEMP(NVEH,NUBR) = XTRACL
PRINT TLAY, MINE NØ., ACLDL(L,NM), DID NØT RELEASE.
PRINT TLAY, MINE NØ., ACLDL(L, NM), LAYED.
MLFLAG=1
MLFLAG=0
35
CØNTINUE


ENTRY
M1 = ACLDL(L, NM)
MT2 = MDL(M1, 2)
CALL RAN1 (IR, RN)
ACMCL(MT2,2) - RN ≥ 0
No
47
Yes
ECCENT = √((1.0 - (ACMCL(MT2,3) - 1.0)**2 - 1000**2) / (1.0 - (ACMCL(MT2,3) - 1.0)**2))
CEP = (√(1.0 - ((1.0 - ACCL(J,4)**2) / (1.0 - ECCENT**2 + 1.0)))) / 3.0
SIGMA = CEP / 1.1744
CALL RAN1 (IR, RN)
48

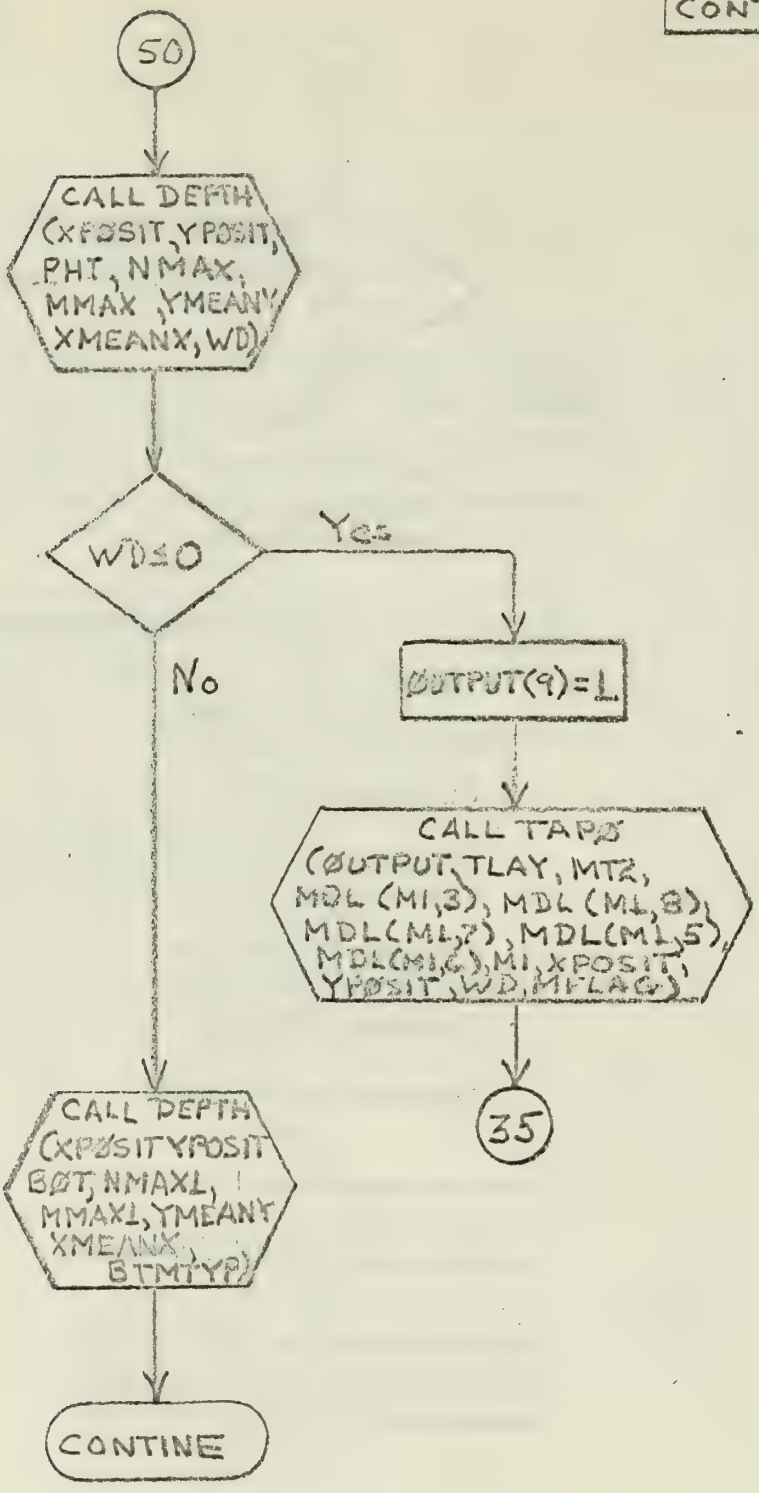
7.3 CONTD

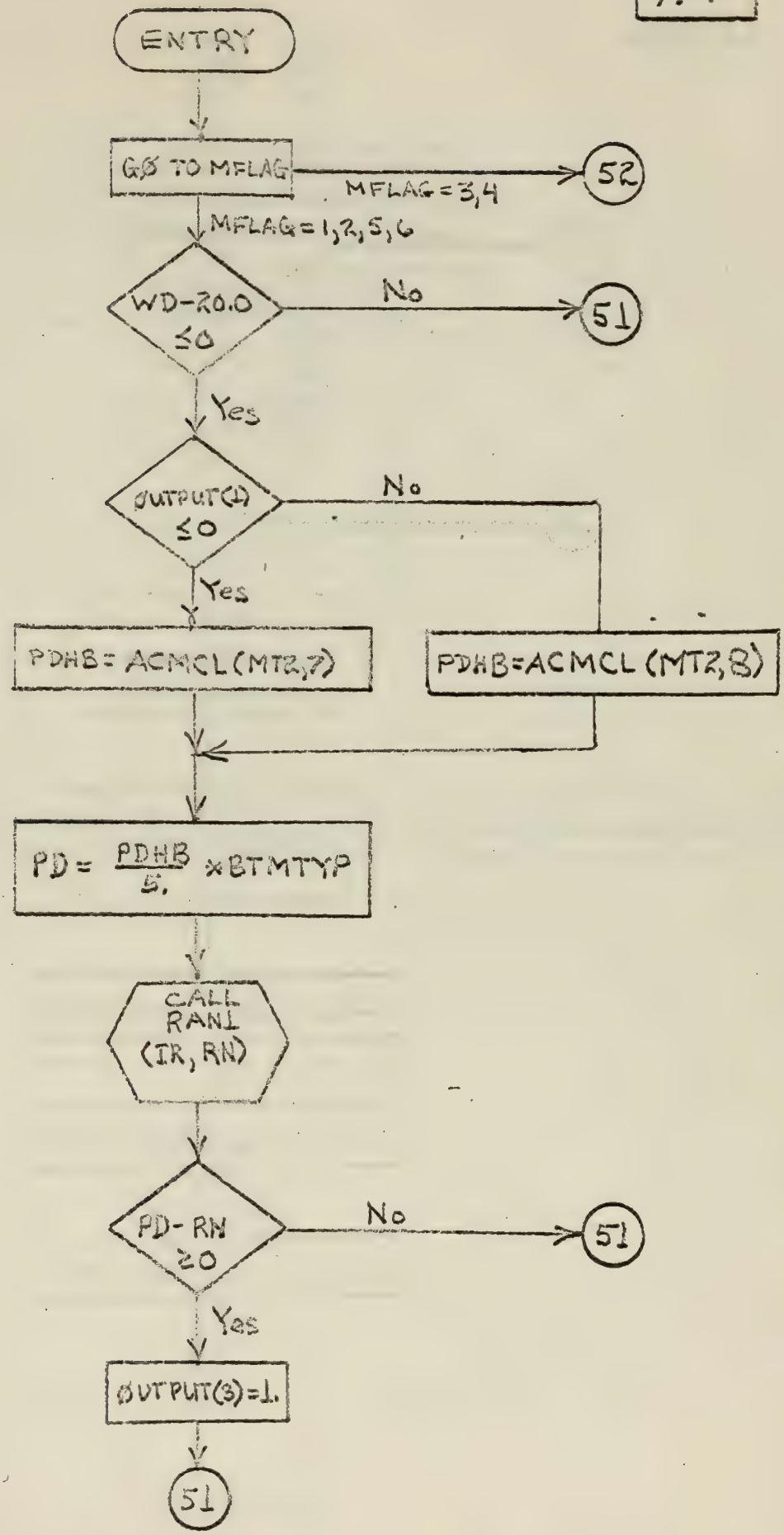

47
OUTPUT(1)=1.
ECCENT = SQRT( 1.0-(ACMCL(MT2,5)-1.0)**2-1000.**2 / 1.0-(ACMCL(MT2,3)-1.0)**2 )
CEP = SQRT( 1.0- ( 1.0 - ACCL(I,4)**2 / 1.0 - ECCENT**2 + 1.0 ) ) / 3.0
SIGMA = CEP / 1.1774
CALL RANI (IR,RN)
ACMCL(MT2,6) - RN ≤ 0
Yes
49
No
OUTPUT(2)=1.
49

7.3 CONT'D

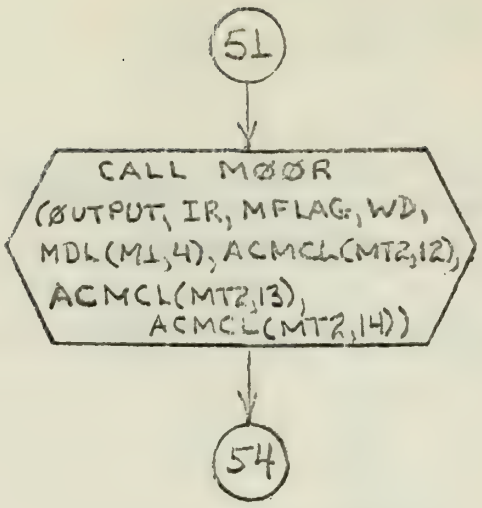

48
ACMCL(MT2,5) - RN ≤0
Yes
49
No
OUTPUT(2)=1.
49
CALL RNNORM (SIGMA, XE, IR)
CALL RNNORM (SIGMA, YE, IR)
XPOSIT = ACLDL(L,NM+1)+XE
YPOSIT = ACLDL(L,NM+2)+YE
MFLAG = ACMCL(MT2,10)
50

7.3 CONTD

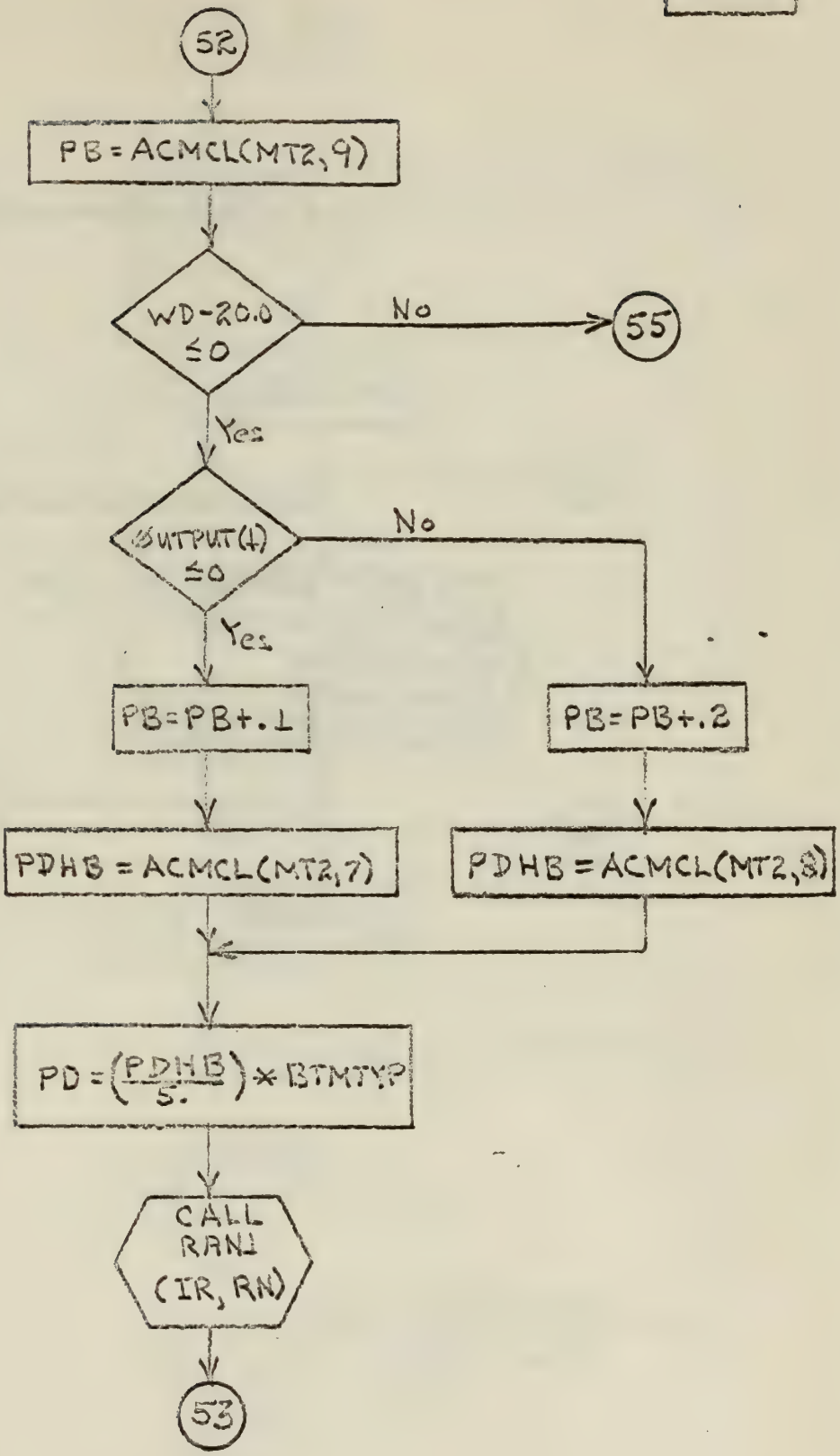

50
CALL DEPTH (XPOSIT, YPOSIT, PHI, NMAX, MMAX, YMEANY, XMEANX, WD)
WD≤0
Yes
No
OUTPUT(9)=1
CALL TAPO (OUTPUT, TLAY, MTZ, MDL(MI,3), MDL(MI,8), MDL(MI,7), MDL(MI,5), MDL(MI,6), MI, XPOSIT, YPOSIT, WD, MFLAG)
35
CALL DEPTH (XPOSITYPOSIT BOT, NMAXL, MMAXL, YMEANY XMEANX, BTMTYP)
CONTINE


ENTRY
GØ TO MFLAG
MFLAG=3,4
52
MFLAG=1,2,5,6
WD-20.0 ≤0
No
51
Yes
OUTPUT(4) ≤0
No
Yes
PDHB=ACMCL(MT2,7)
PDHB=ACMCL(MT2,8)
PD = PDHB / 5. × BTMTYP
CALL RANI (IR, RN)
PD-RN ≥0
No
51
Yes
OUTPUT(3)=1.
51

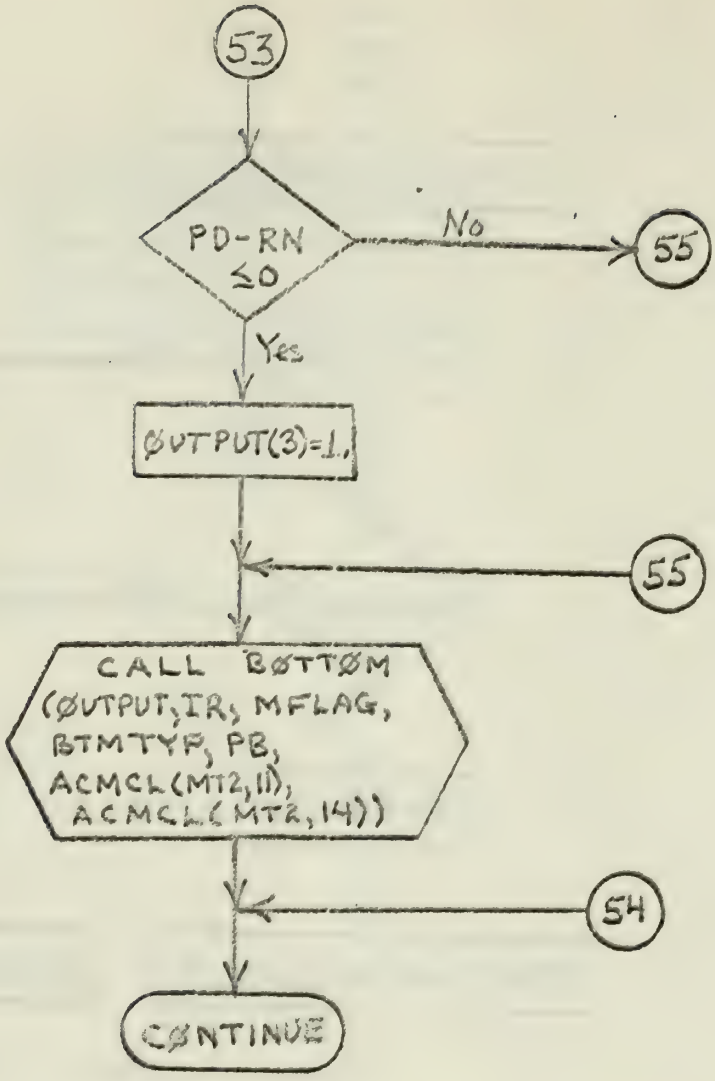
7.4
CONTD

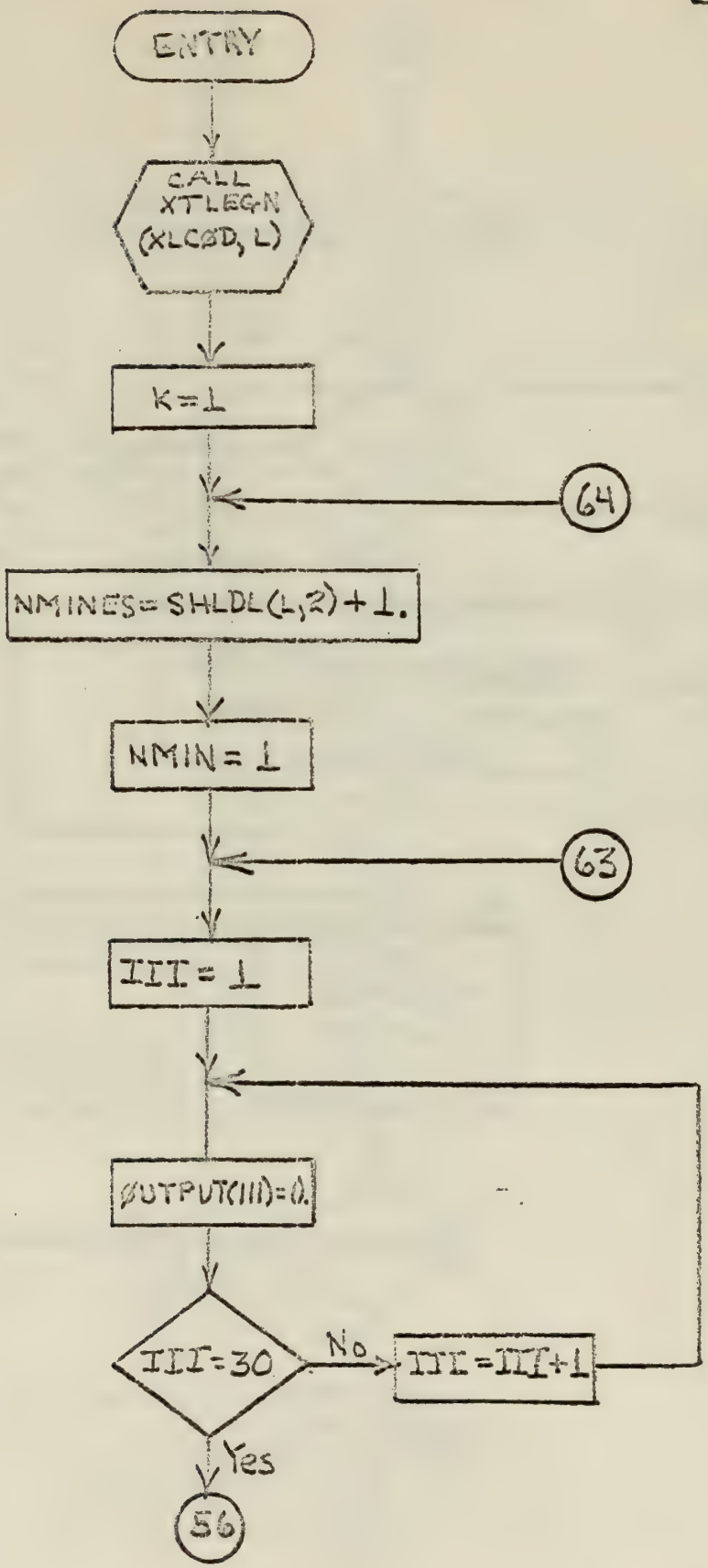

51
CALL MØØR
(ØUTPUT, IR, MFLAG, WD,
MDL(M1,4), ACMCL(MT2,12),
ACMCL(MT2,13),
ACMCL(MT2,14))
54

7.4 CONTD

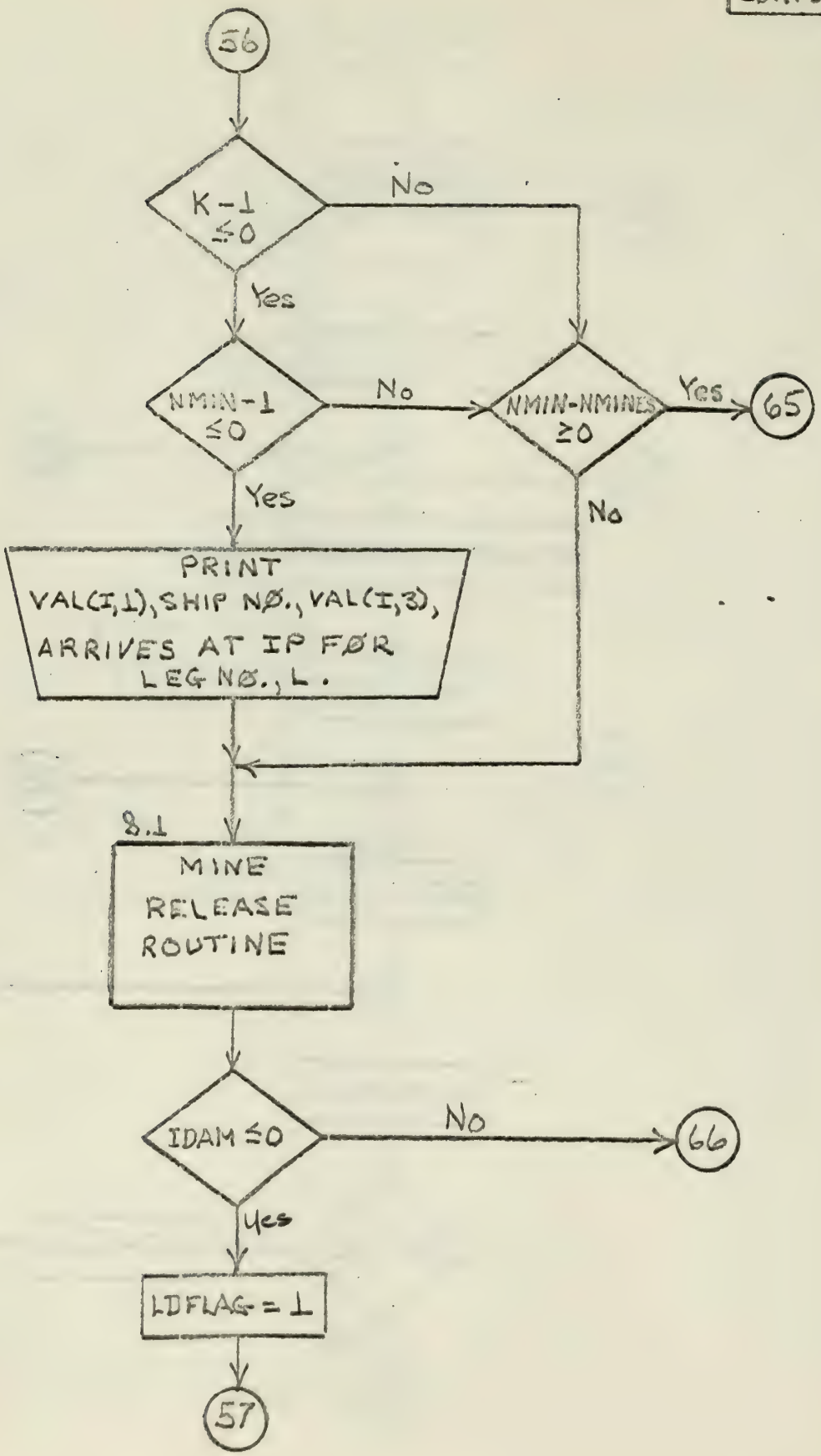

52
PB = ACMCL(MT2, 9)
WD - 20.0 ≤ 0
No
55
Yes
⌀UTPUT(1) ≤ 0
No
Yes
PB = PB + .1
PB = PB + .2
PDHB = ACMCL(MT2, 7)
PDHB = ACMCL(MT2, 8)
PD = (PDHB / 5.) * BTMTYP
CALL RANI (IR, RN)
53

7.4 CONTD

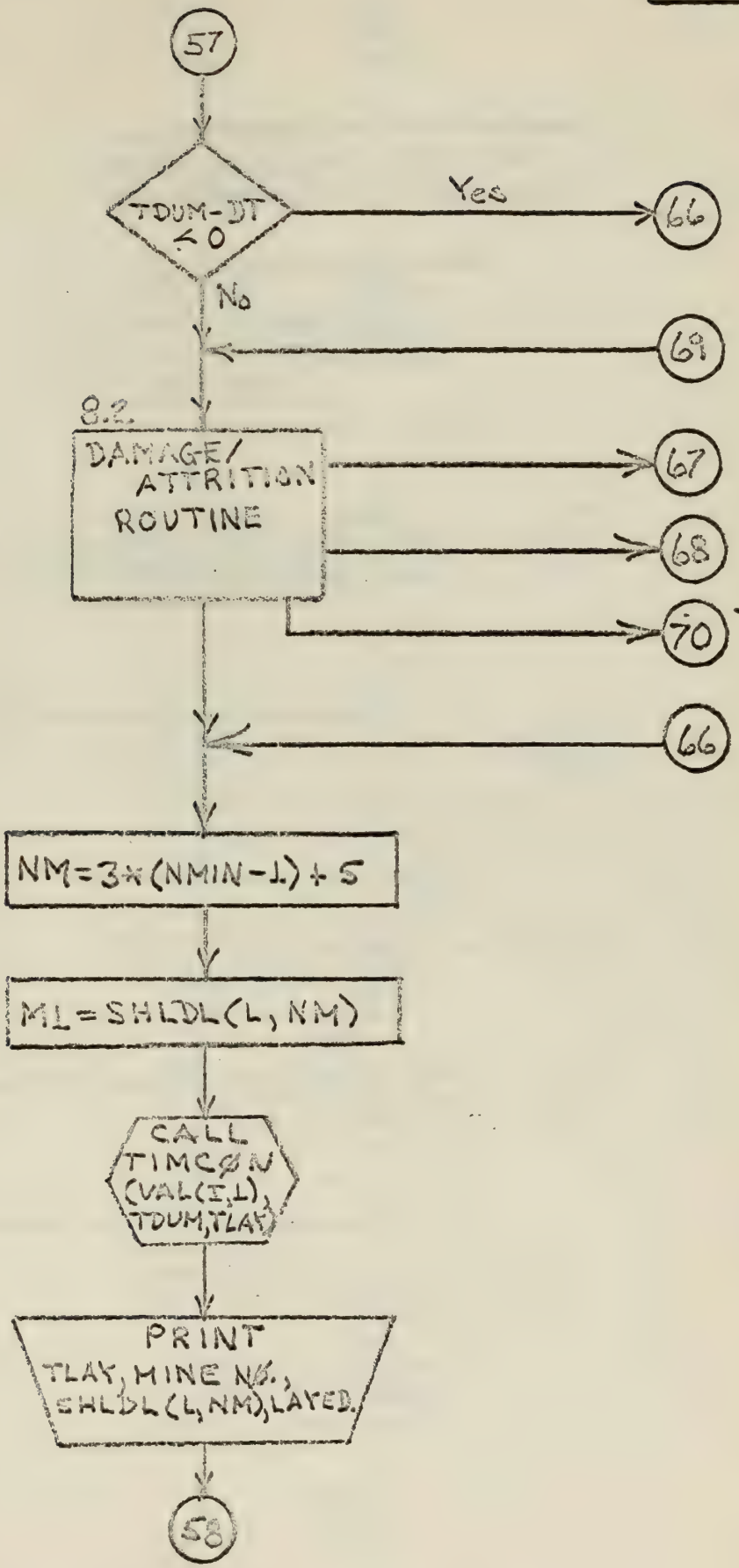

53
PD-RN ≤0
No
55
Yes
ØUTPUT(3)=1.
55
CALL BØTTØM (ØUTPUT,IR, MFLAG, BTMTYP, PB, ACMCL(MT2,11), ACMCL(MT2,14))
54
CØNTINUE


ENTRY
CALL XTLEGN (XLCBD, L)
K=1
64
NMINES=SHLDL(L,2)+1.
NMIN=1
63
III=1
OUTPUT(III)=0.
III=30
No
III=III+1
Yes
56

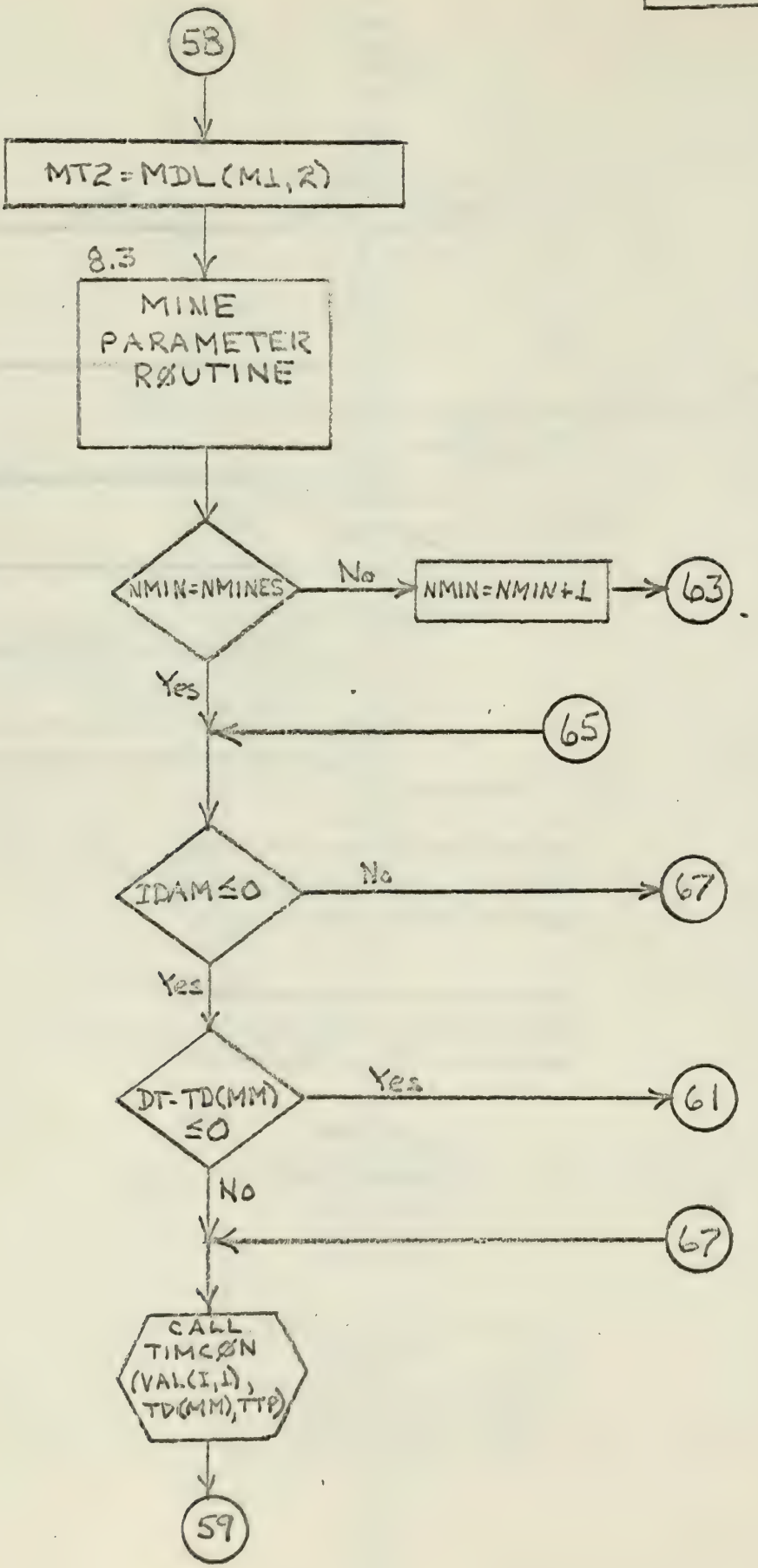
8.
CONT'D


56
K-1 ≤0
No
Yes
NMIN-1 ≤0
No
NMIN-NMINES ≥0
Yes
65
Yes
No
PRINT VAL(I,1), SHIP NØ., VAL(I,3), ARRIVES AT IP FØR LEG NØ., L.
8.1
MINE RELEASE ROUTINE
IDAM ≤0
No
66
Yes
LDFLAG = 1
57

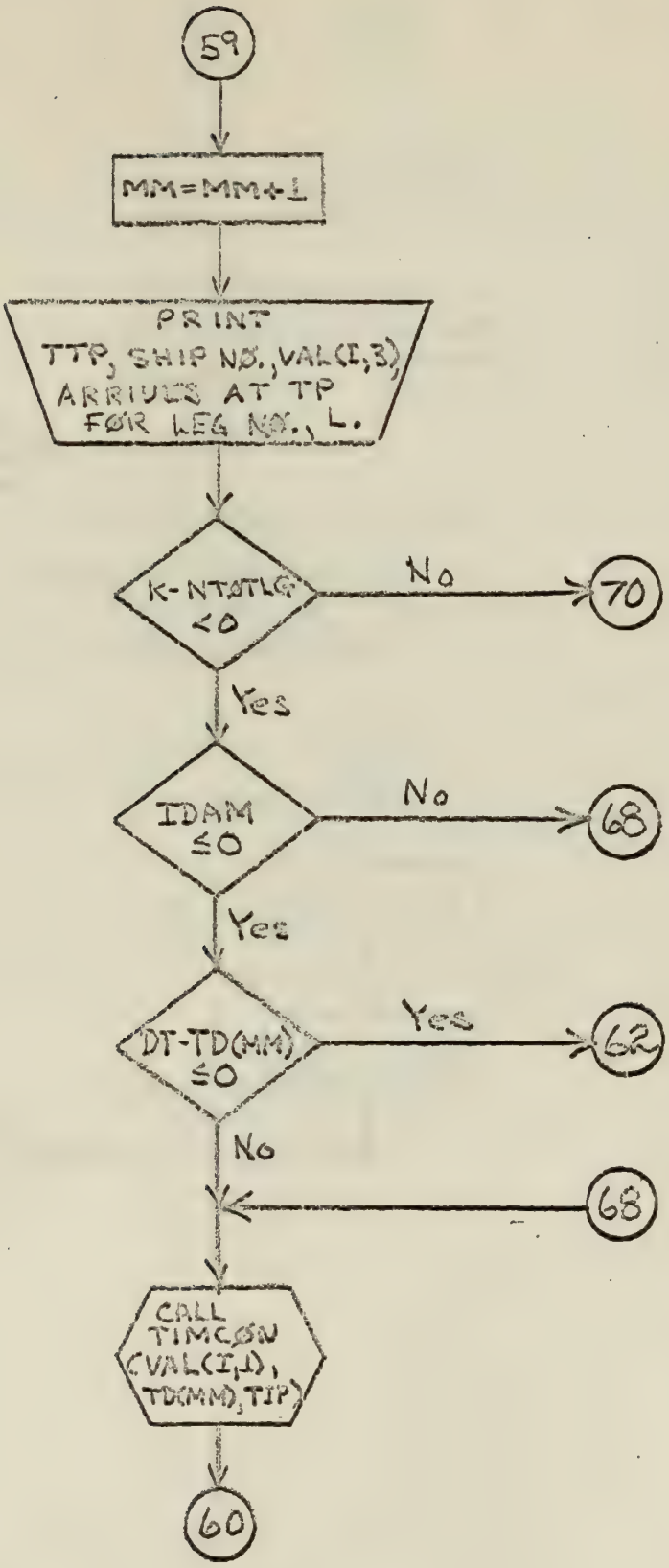
8.
CONTD


57
TDUM-DT
< 0
Yes
66
No
69
8.2.
DAMAGE/
ATTRITION
ROUTINE
67
68
70
66
NM=3*(NMIN-1)+5
ML=SHLDL(L,NM)
CALL
TIMCON
(VAL(I,L),
TDUM,TLAY)
PRINT
TLAY, MINE NO.,
SHLDL(L,NM), LAYER
58

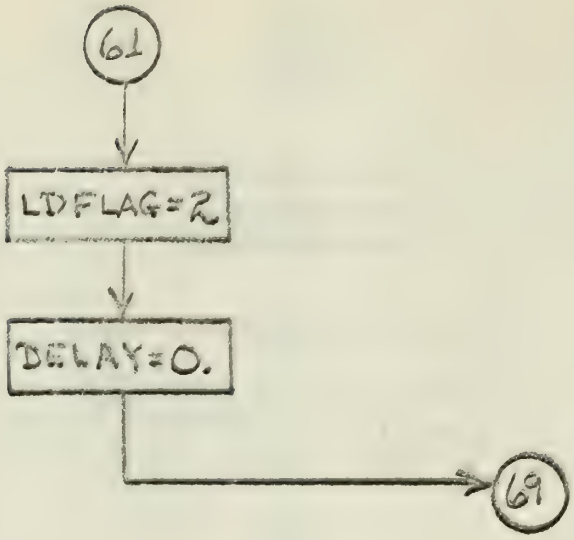
8
CONTD

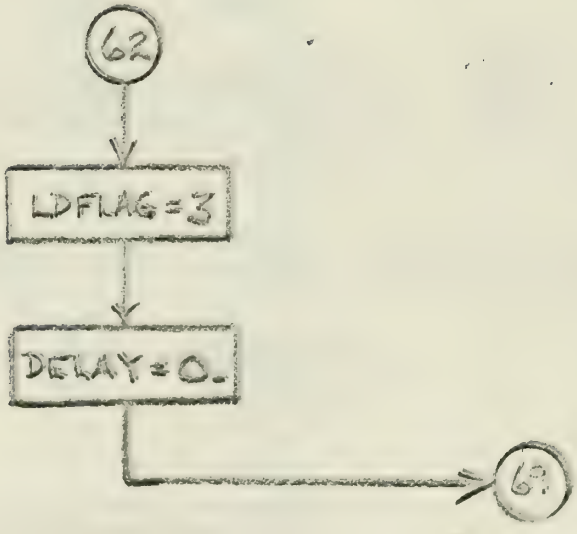

58
MT2 = MDL(M1,2)
8.3
MINE PARAMETER ROUTINE
NMIN=NMINES
No
NMIN=NMIN+1
63
Yes
65
IDAM≤0
No
67
Yes
DT-TD(MM) ≤0
Yes
61
No
67
CALL TIMCON (VAL(I,1), TD(MM), TTP)
59

8.
CONTD

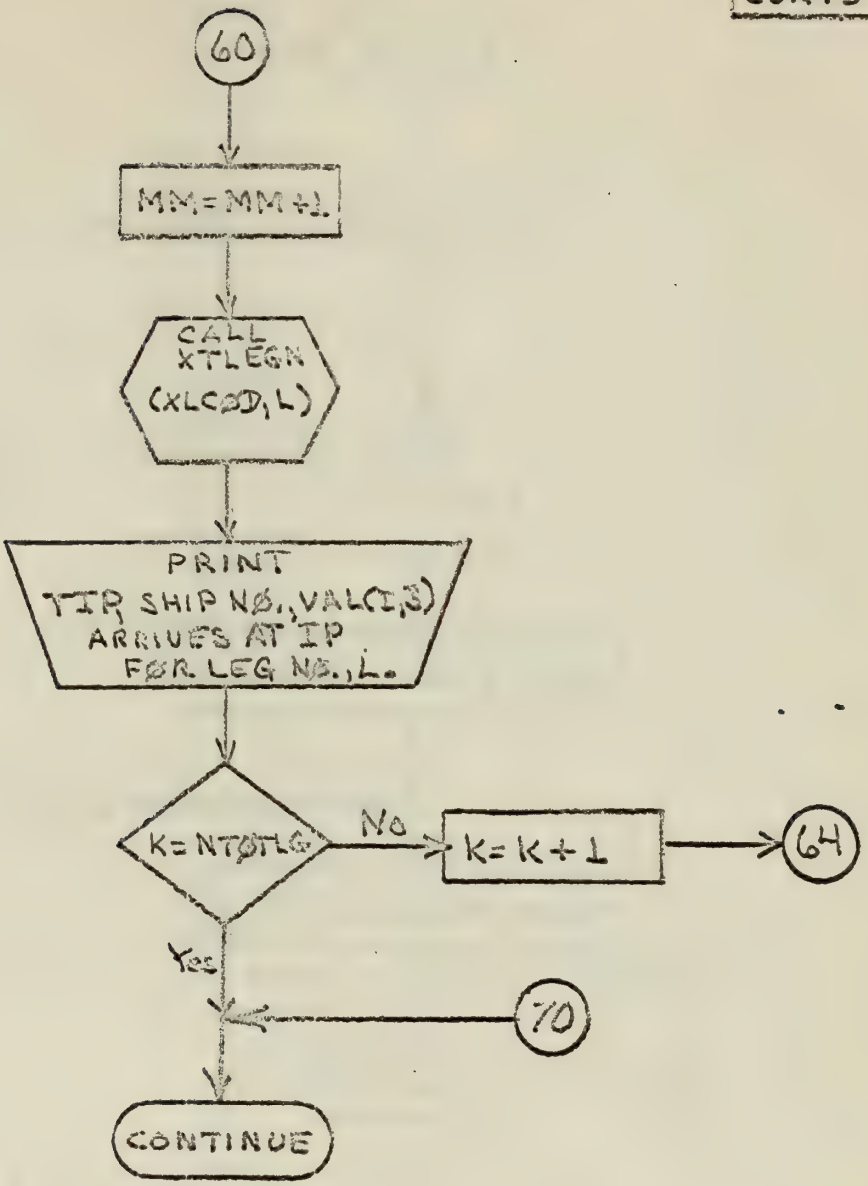

59
MM=MM+1
PRINT
TTP, SHIP NO., VAL(I,3),
ARRIVES AT TP
FOR LEG NO., L.
K-NTOTLG
<0
No
70
Yes
IDAM
≤0
No
68
Yes
DT-TD(MM)
≤0
Yes
62
No
68
CALL
TIMCON
(VAL(I,J),
TD(MM),TIP)
60

8. CONTD

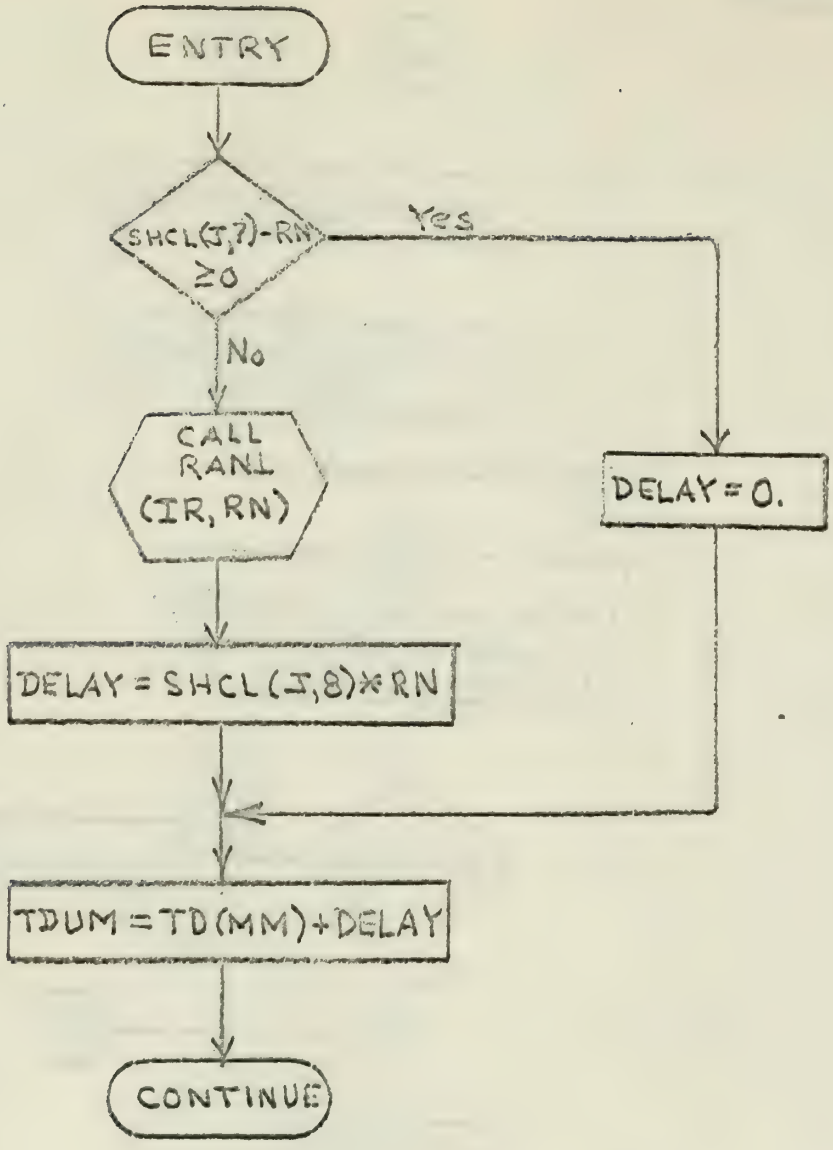


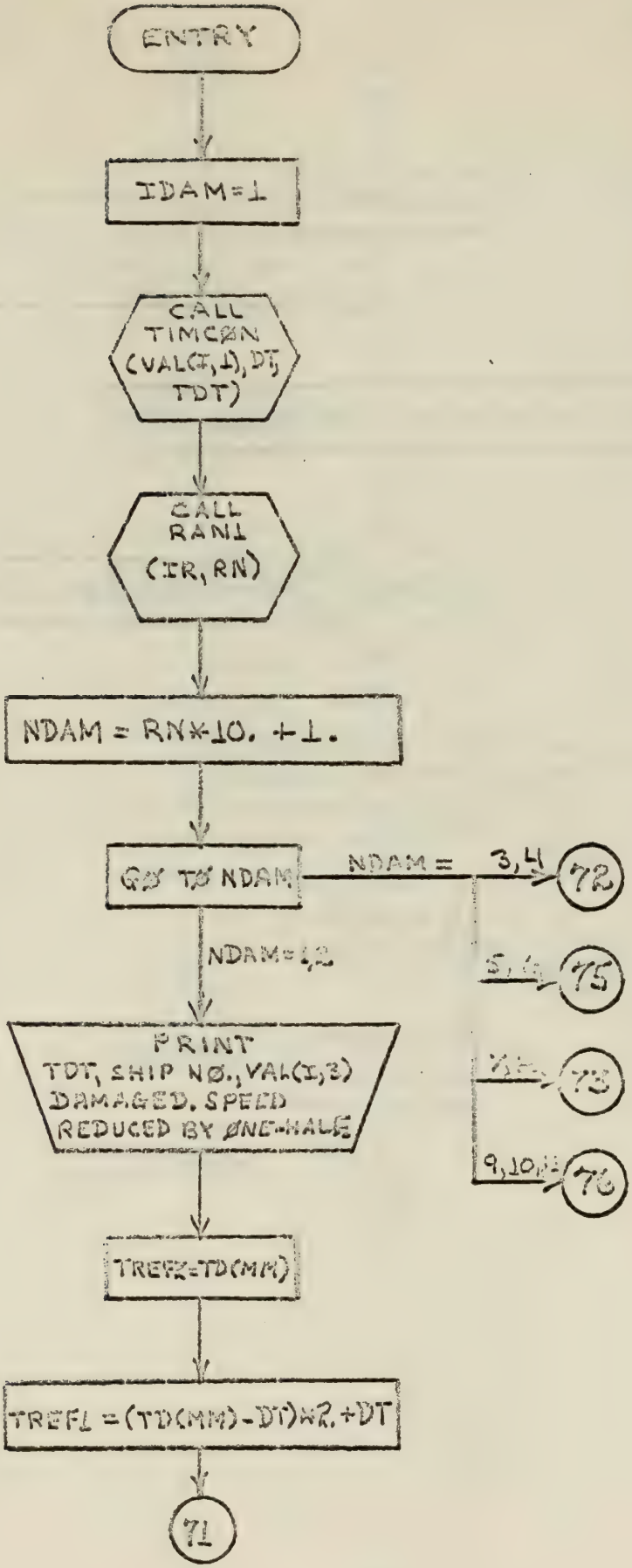



8. CONTD

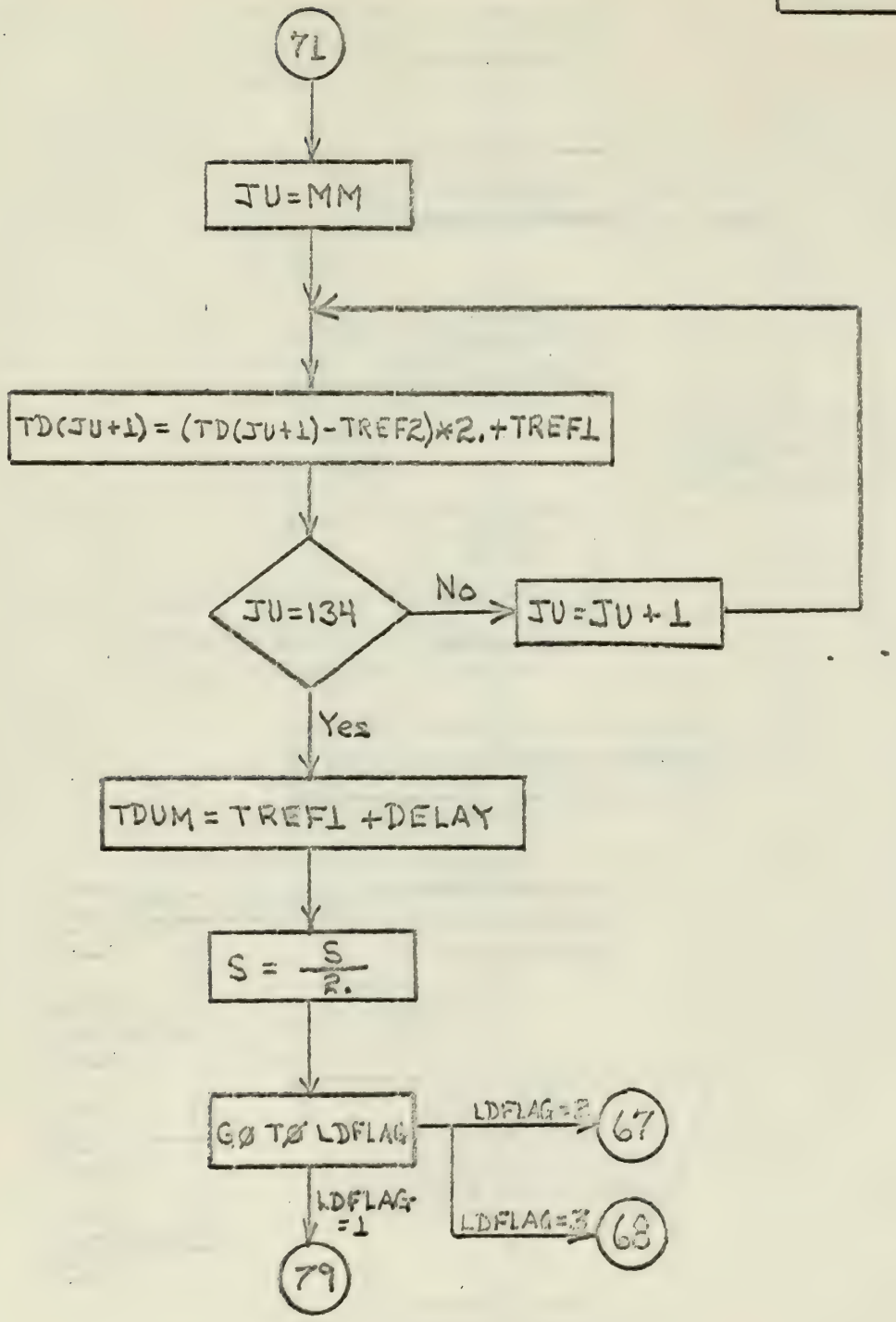

60
MM=MM+1.
CALL XTLEGN (XLCØD,L)
PRINT TIP, SHIP NØ.,VAL(I,3) ARRIVES AT IP FØR. LEG NØ.,L.
K=NTØTLG
No
K=K+1
64
Yes
70
CONTINUE

8.1

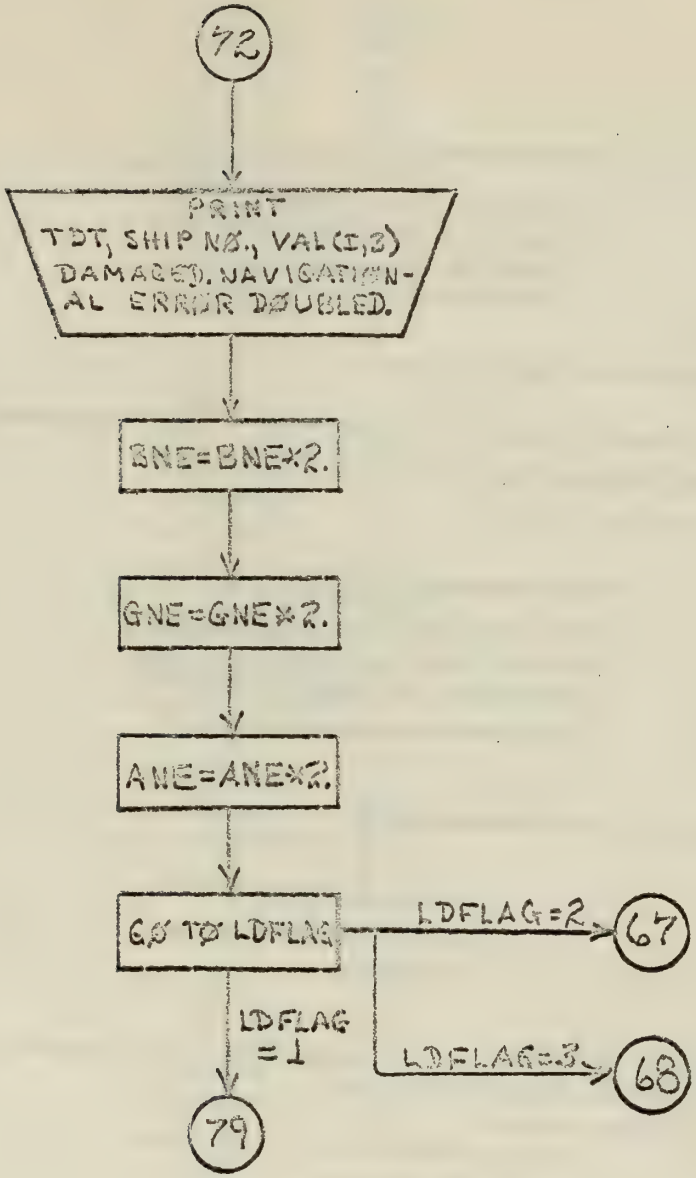

ENTRY
SHCL(I,7)-RN ≥0
Yes
No
CALL RANL (IR, RN)
DELAY = 0.
DELAY = SHCL(I,8)*RN
TDUM = TD(MM)+DELAY
CONTINUE


ENTRY
IDAM=1
CALL TIMCØN (VAL(I,1), DT, TDT)
CALL RAN1 (IR, RN)
NDAM = RN*10. +1.
GØ TØ NDAM
NDAM = 3,4
72
NDAM=1,2
5,6
75
PRINT TDT, SHIP NØ., VAL(I,3) DAMAGED. SPEED REDUCED BY ØNE-HALF
7,8
73
9,10
76
TREFL=TD(MM)
TREFL = (TD(MM)-DT)*R. +DT
71

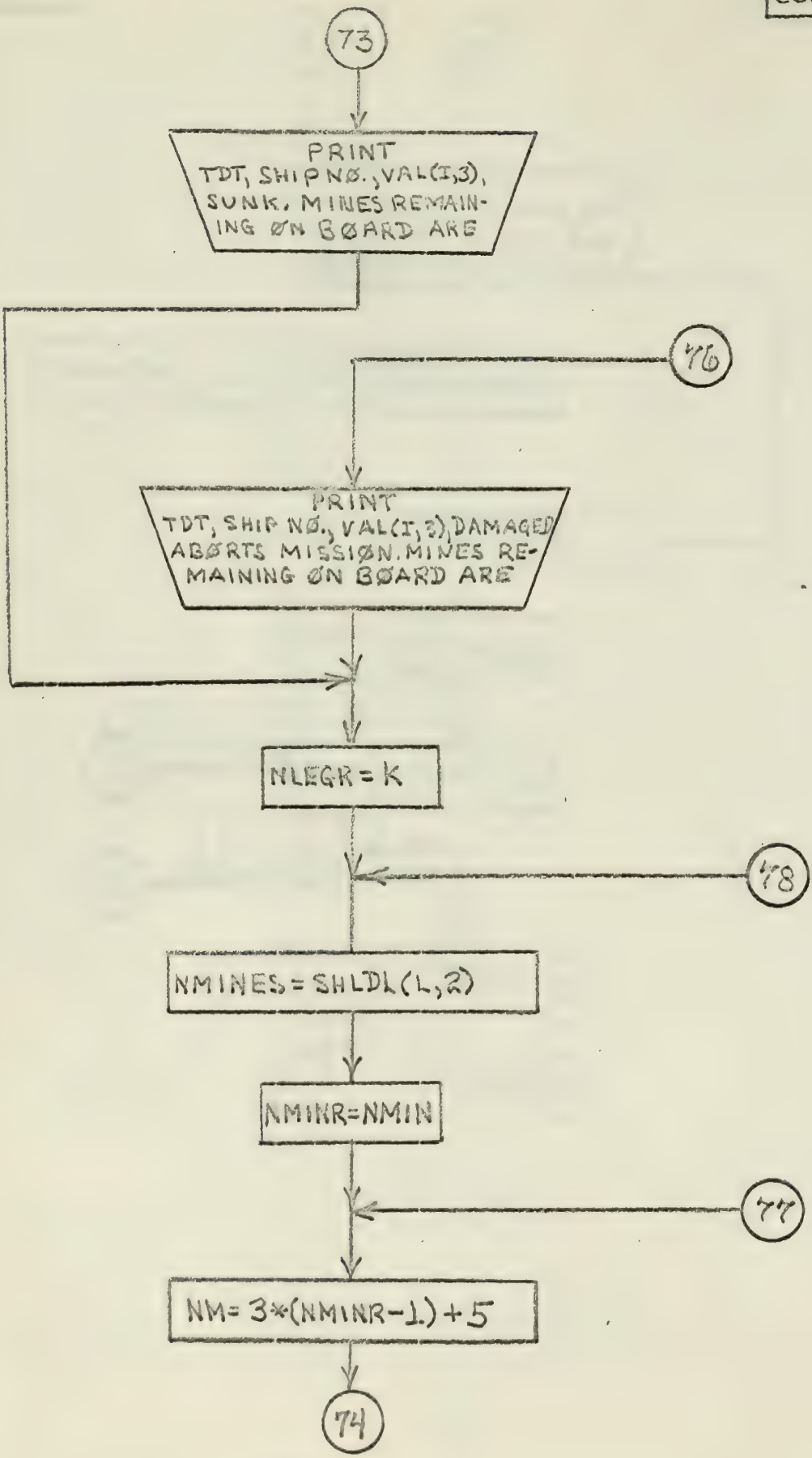
8.2
CONTD


71
JU=MM
TD(JU+1) = (TD(JU+1)-TREF2)*2.+TREF1
JU=134
No
JU=JU+1
Yes
TDUM = TREF1 +DELAY
S = S/2.
GØ TØ LDFLAG
LDFLAG=2
67
LDFLAG=3
68
LDFLAG=1
79

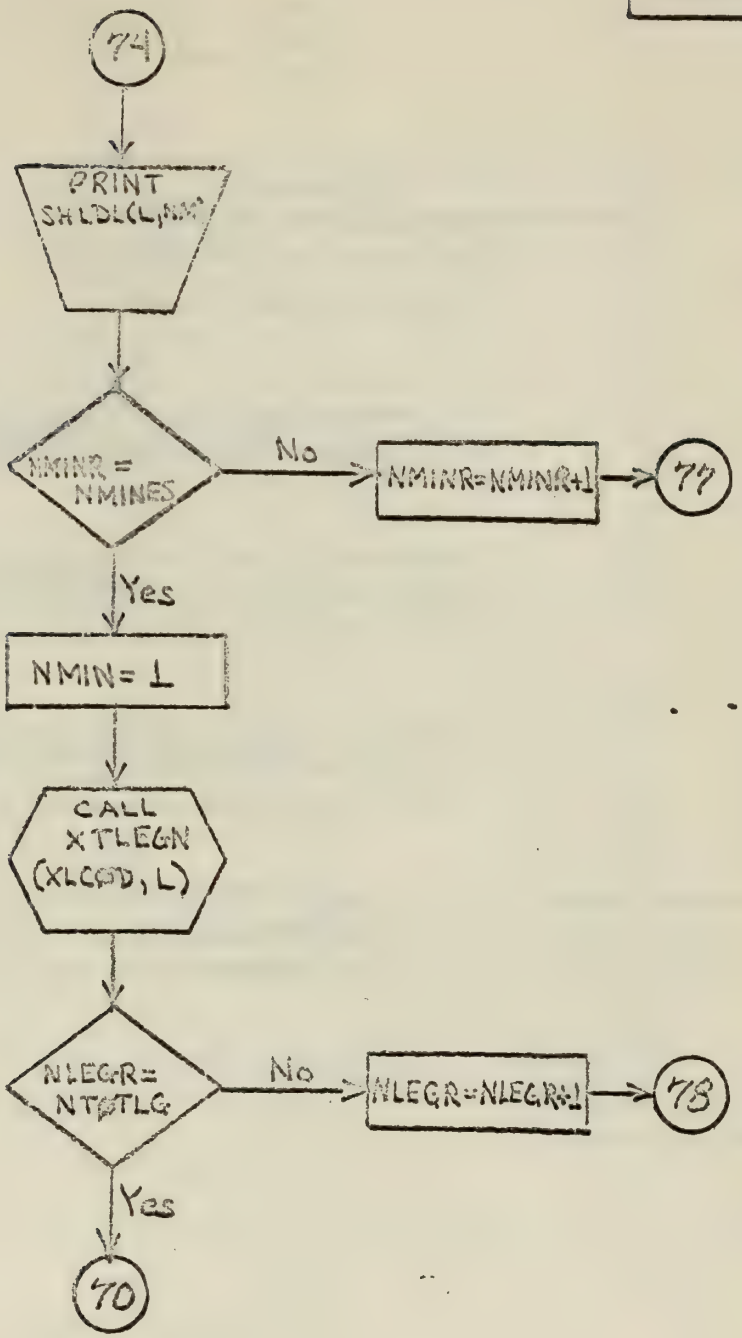
8.2
CONT'D


72
PRINT
TDT, SHIP NØ., VAL(I,3)
DAMAGED. NAVIGATION-
AL ERROR DØUBLED.
BNE=BNE*2.
GNE=GNE*2.
ANE=ANE*2.
GØ TØ LDFLAG
LDFLAG=2
67
LDFLAG
=1
LDFLAG=3
68
79

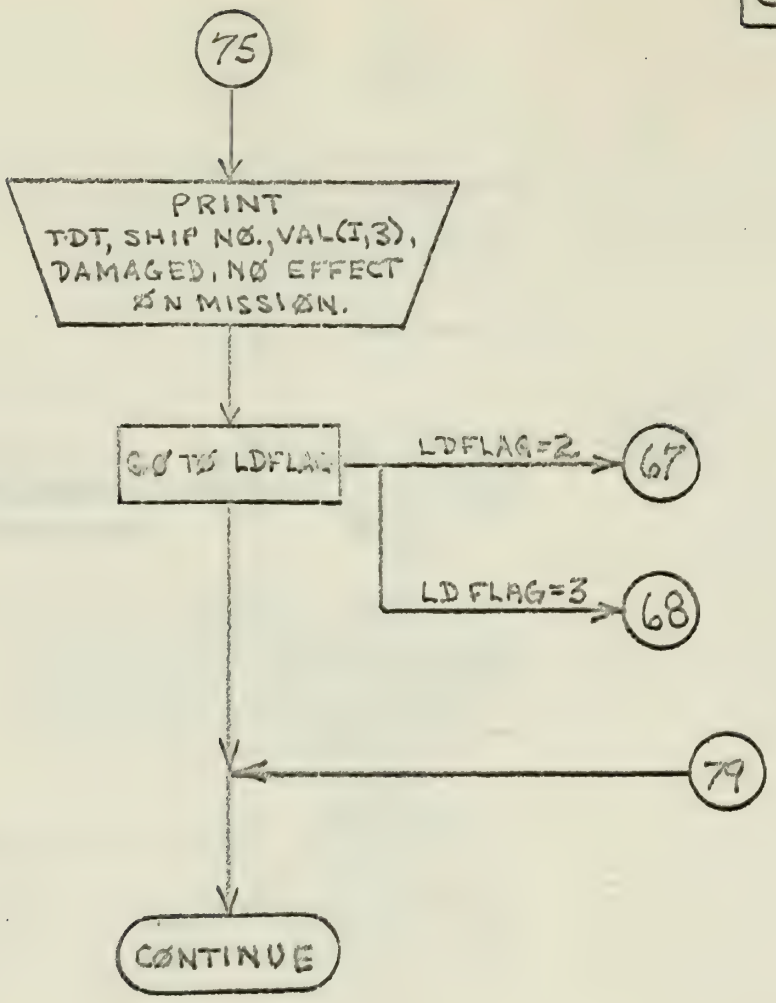
8.2 CONT'D

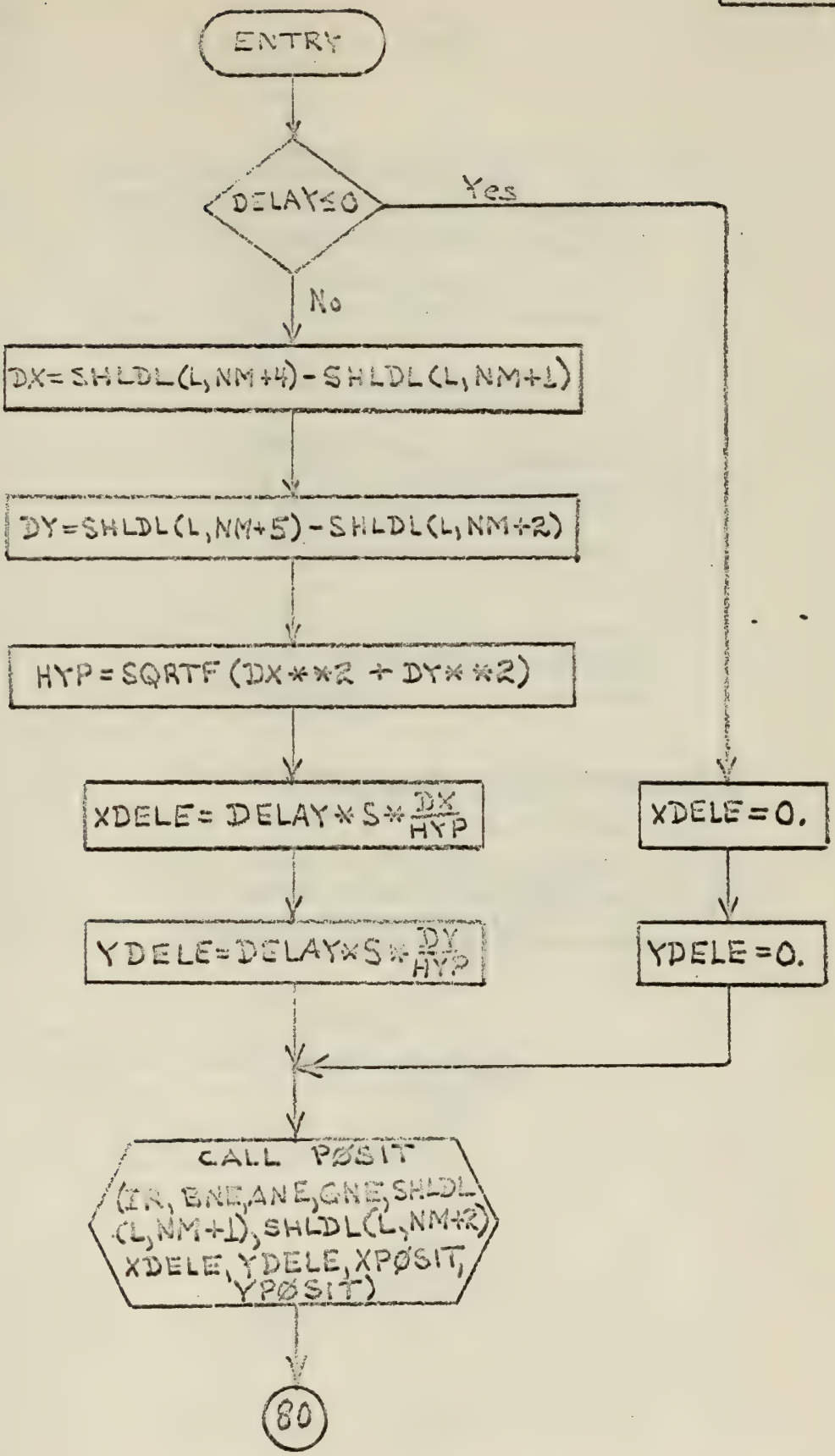

73
PRINT TDT, SHIP NØ., VAL(I,3), SUNK, MINES REMAINING ØN BØARD ARE
76
PRINT TDT, SHIP NØ., VAL(I,3), DAMAGED ABØRTS MISSIØN, MINES REMAINING ØN BØARD ARE
NLEGR = K
78
NMINES = SHLDL(L,2)
NMINR=NMIN
77
NM= 3*(NMINR-1.)+5
74

8.2 CONTD

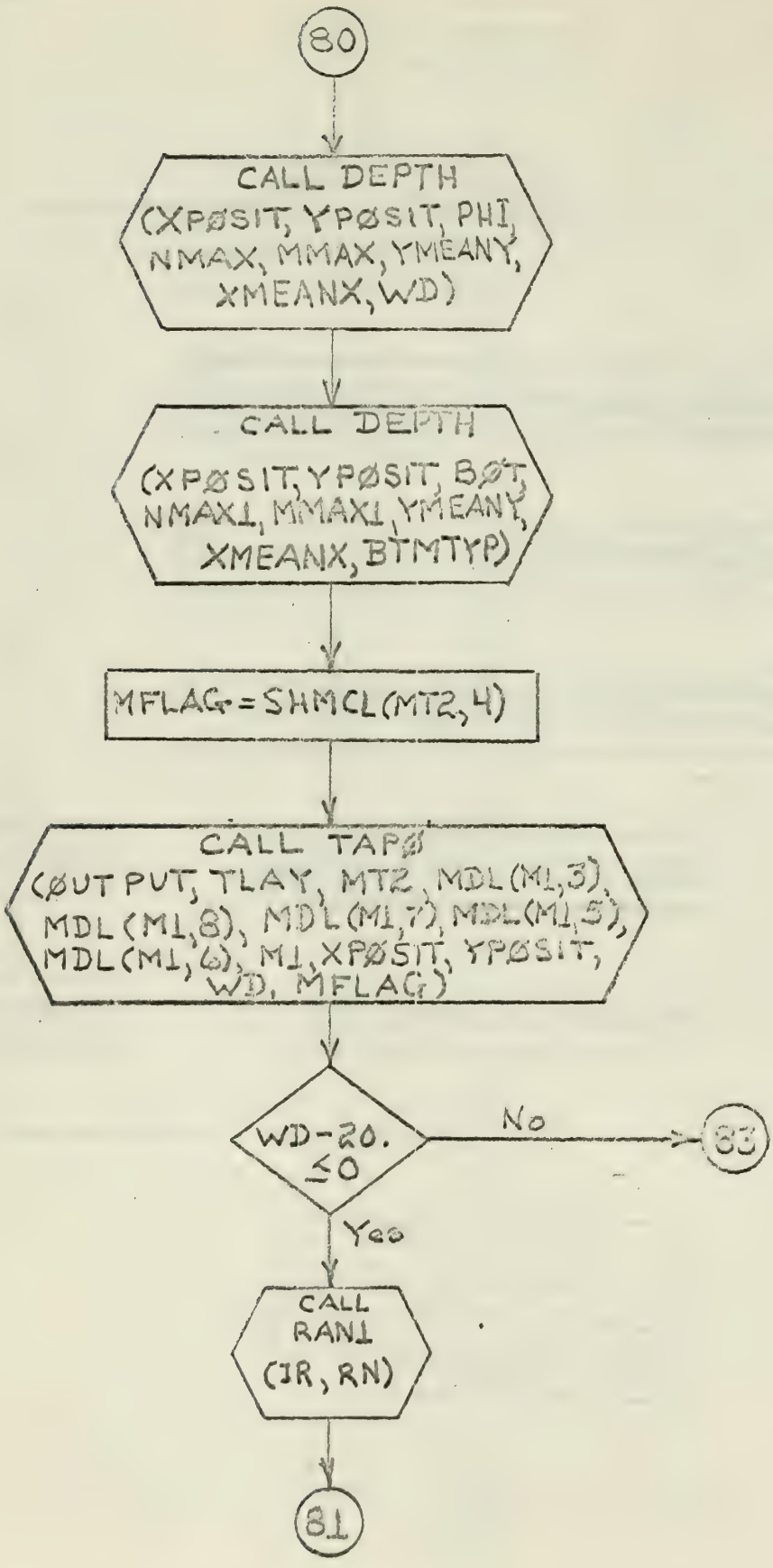

74
PRINT SHLDL(L),NM
NMINR = NMINES
No
NMINR=NMINR+1
77
Yes
NMIN=1
CALL XTLEGN (XLCOD, L)
NLEGR = NTOTLG
No
NLEGR=NLEGR+1
78
Yes
70

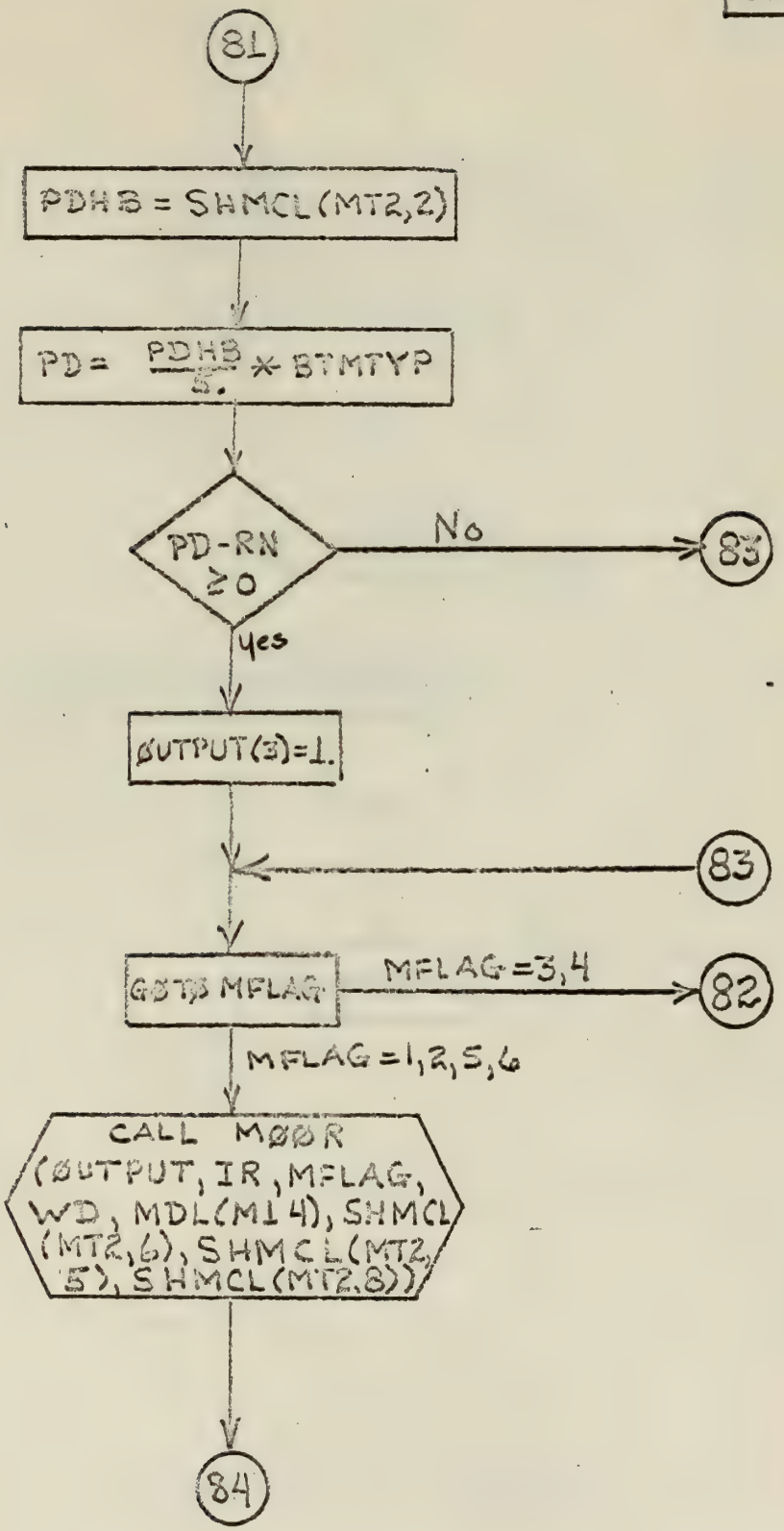
8.2
CONTD


75
PRINT
TDT, SHIP NO., VAL(I,3),
DAMAGED, NO EFFECT
ON MISSION.
GO TO LDFLAG
LDFLAG=2
67
LD FLAG=3
68
79
CONTINUE

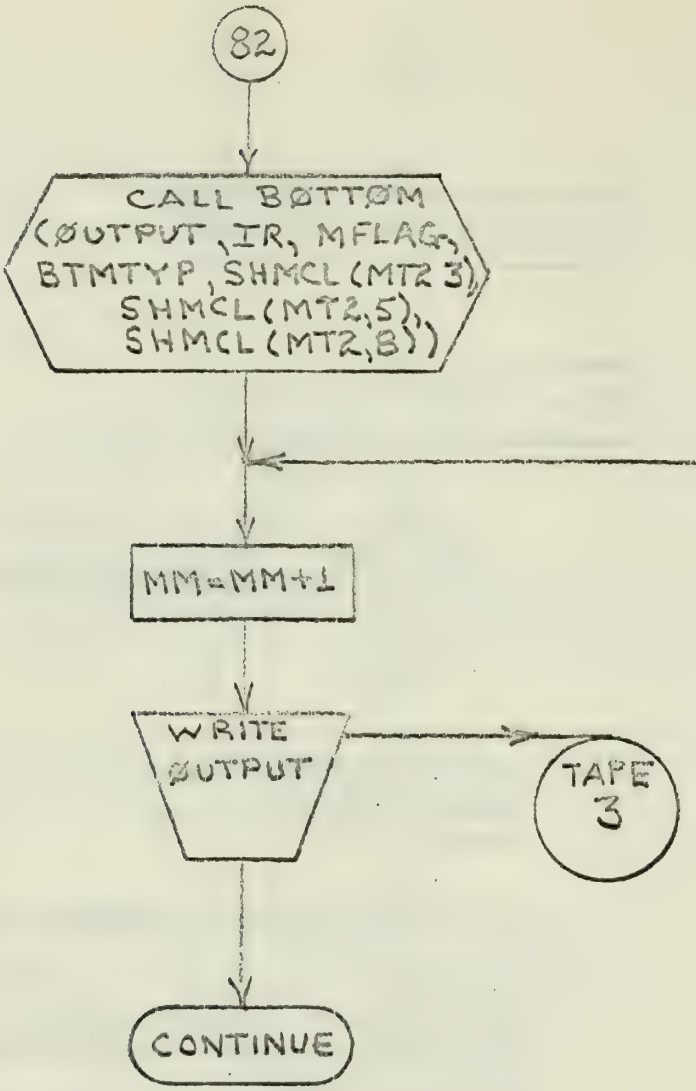

ENTRY
DELAY≤0
Yes
No
DX=SHLDL(L,NM+4)-SHLDL(L,NM+1)
DY=SHLDL(L,NM+5)-SHLDL(L,NM+2)
HYP=SQRTF(DX**2+DY**2)
XDELE=DELAY*S*DX/HYP
YDELE=DELAY*S*DY/HYP
XDELE=0.
YDELE=0.
CALL PØSIT
(I,R,BNE,ANE,GNE,SHLDL
(L,NM+1),SHLDL(L,NM+2)
XDELE,YDELE,XPØSIT,
YPØSIT)
80

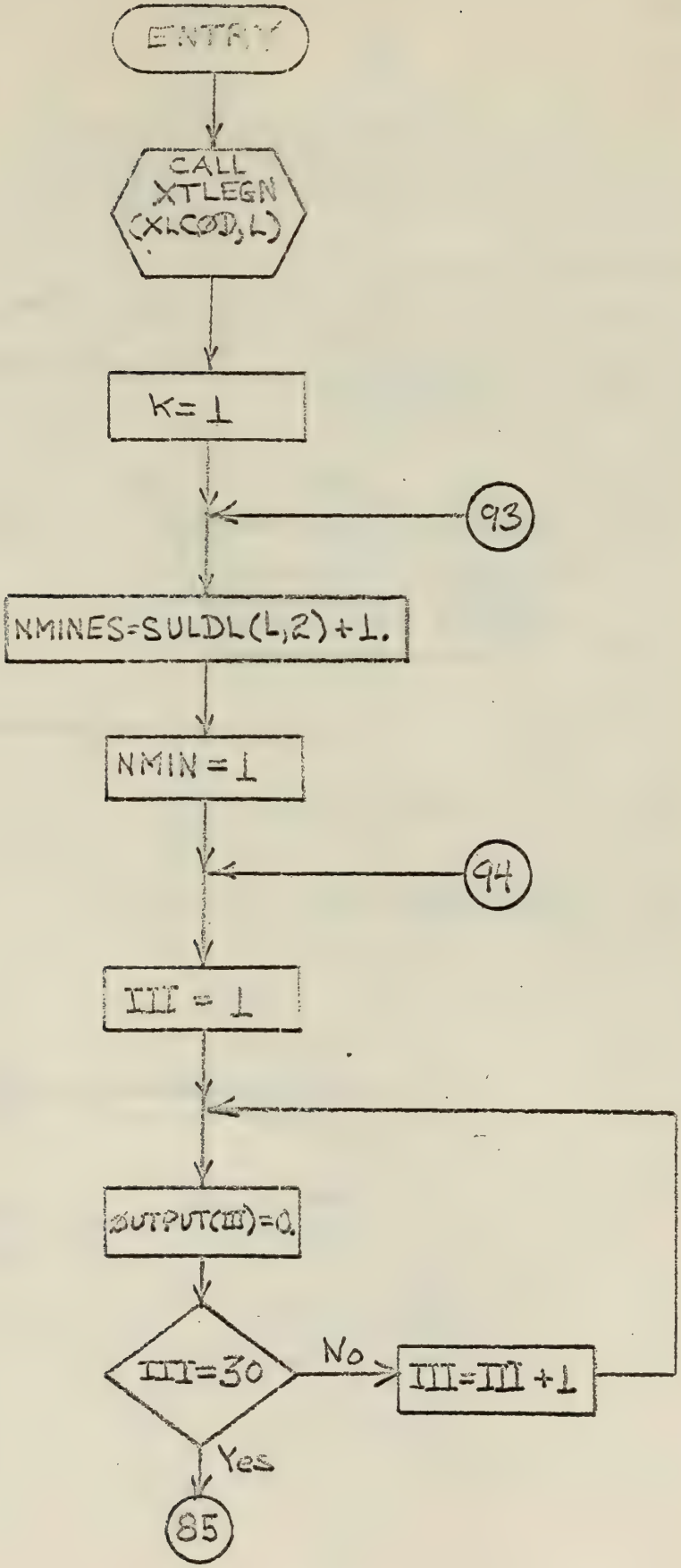
8.3
CONTD

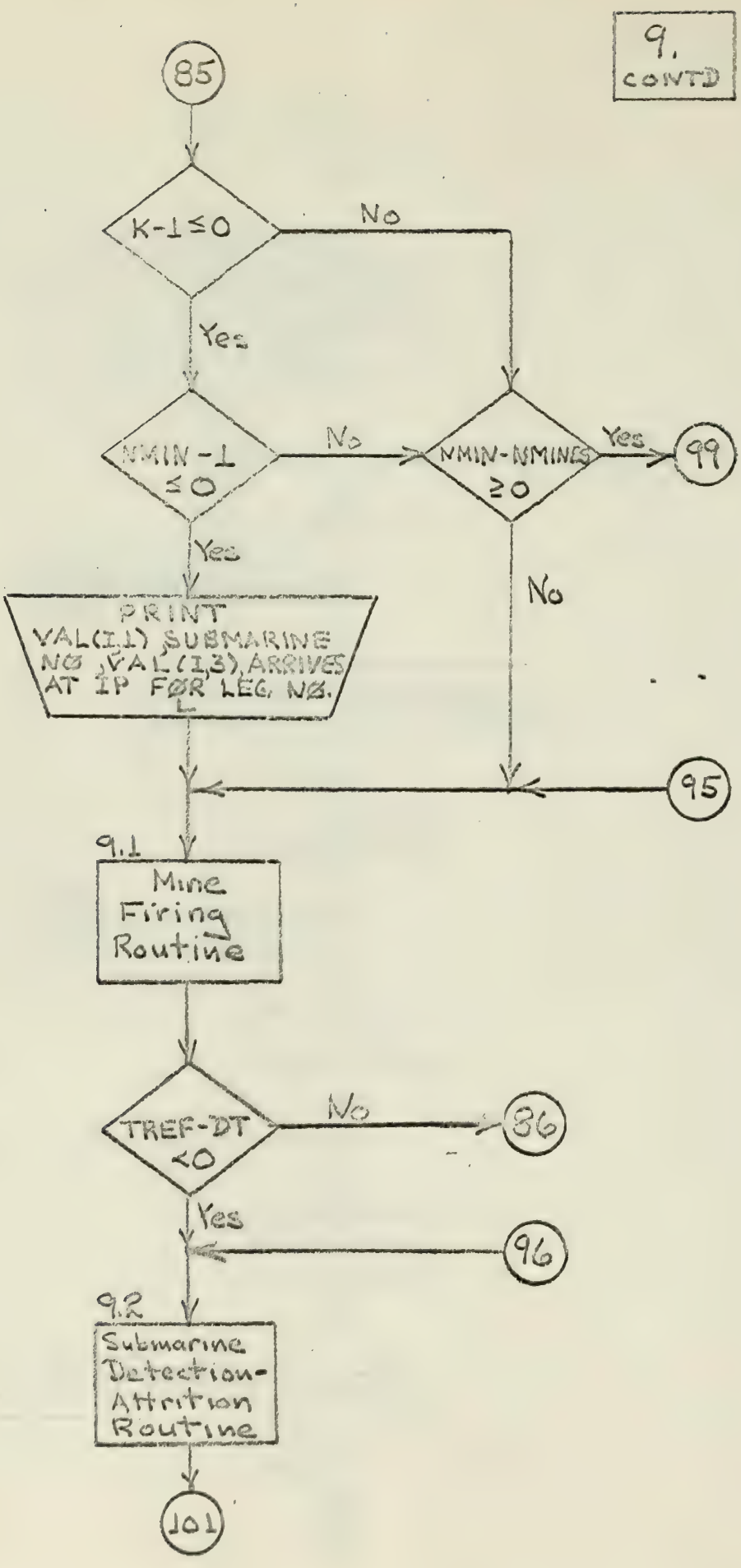

80
CALL DEPTH
(XPØSIT, YPØSIT, PHI, NMAX, MMAX, YMEANY, XMEANX, WD)
CALL DEPTH
(XPØSIT, YPØSIT, BØT, NMAX1, MMAX1, YMEANY, XMEANX, BTMTYP)
MFLAG = SHMCL(MT2, 4)
CALL TAPØ
(ØUTPUT, TLAY, MT2, MDL(M1,3), MDL(M1,8), MDL(M1,7), MDL(M1,5), MDL(M1,6), M1, XPØSIT, YPØSIT, WD, MFLAG)
WD-20. ≤0
No
83
Yes
CALL RAN1
(IR, RN)
81


81
PDHB = SHMCL(MT2,2)
PD = PDHB / 5. * BTMTYP
PD-RN ≥ 0
No
83
Yes
OUTPUT(3)=1.
83
GO TO MFLAG
MFLAG=3,4
82
MFLAG=1,2,5,6
CALL MOOR (OUTPUT, IR, MFLAG, WD, MDL(M14), SHMCL(MT2,6), SHMCL(MT2,5), SHMCL(MT2,8))
84

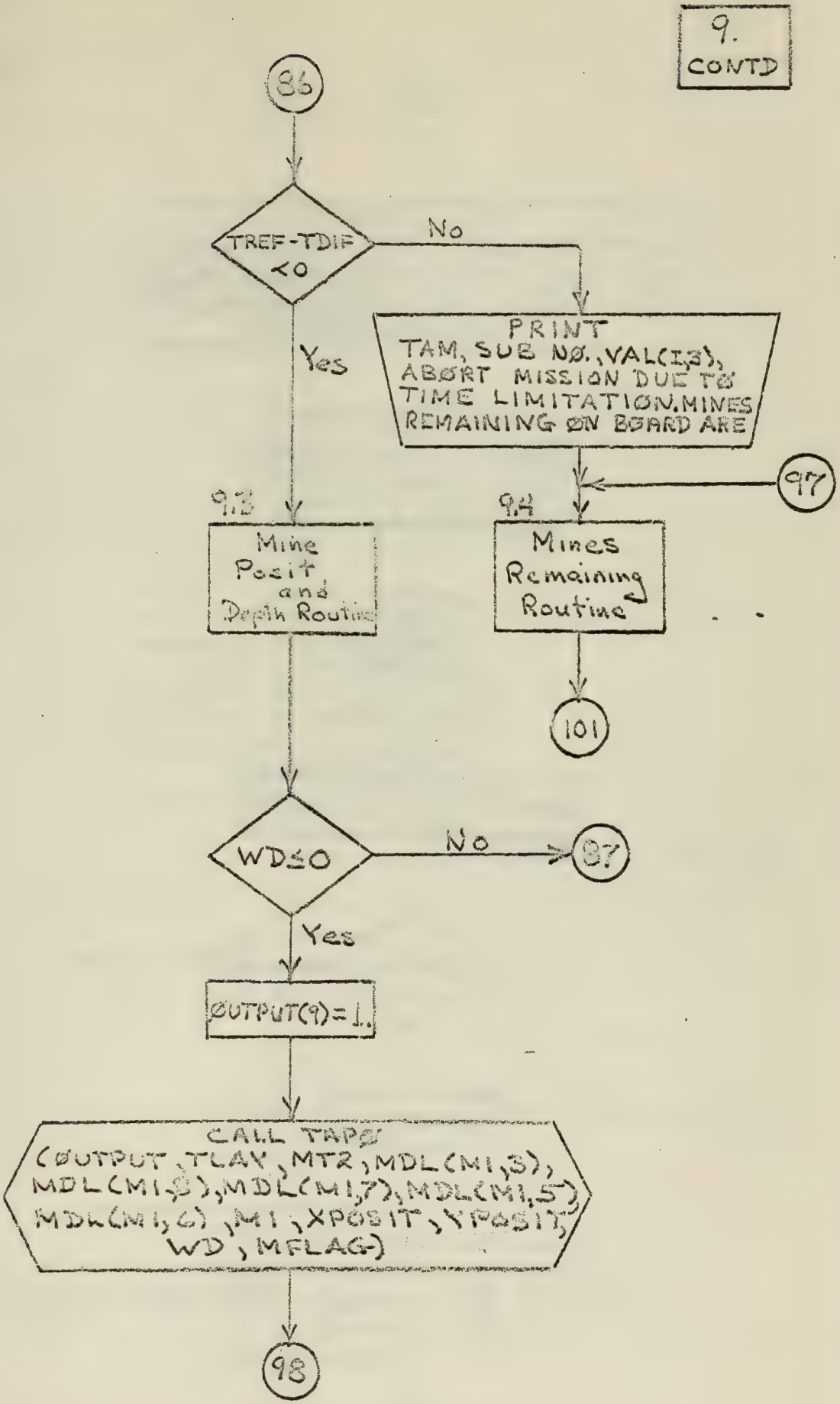
8.3
CONTD


82
CALL BØTTØM
(ØUTPUT, IR, MFLAG,
BTMTYP, SHMCL(MT2,3),
SHMCL(MT2,5),
SHMCL(MT2,8))
84
MM=MM+1
WRITE
ØUTPUT
TAPE
3
CONTINUE

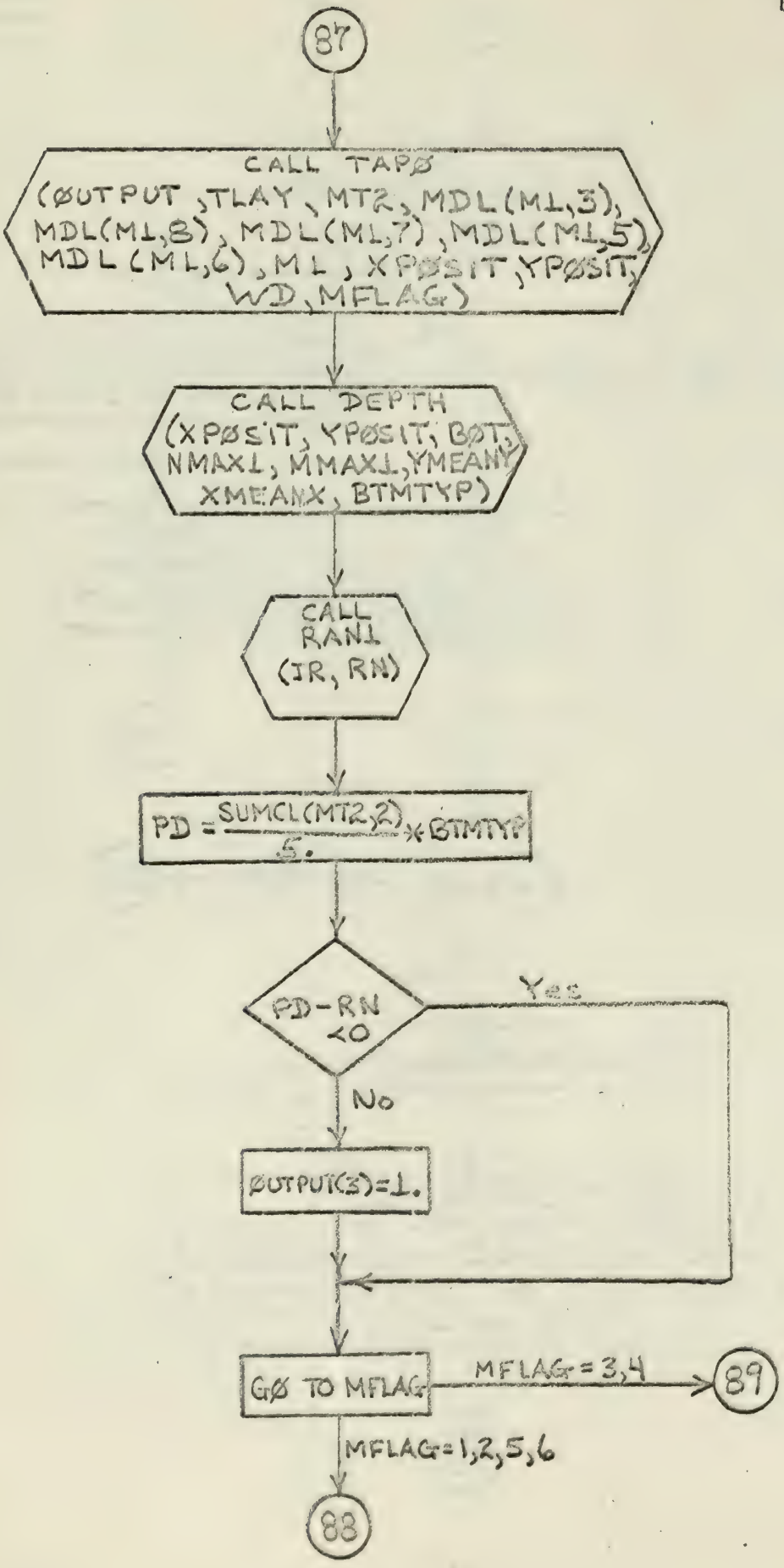

ENTRY
CALL XTLEGN (XLCOD, L)
K=1
93
NMINES=SULDL(L,2)+1.
NMIN=1
94
III=1
OUTPUT(III)=0.
III=30
No
III=III+1
Yes
85


9.
CONTD
85
K-1≤0
No
Yes
NMIN-1
≤0
No
NMIN-NMINES
≥0
Yes
99
Yes
No
PRINT
VAL(I,1) SUBMARINE
NO, VAL(I,3) ARRIVES
AT IP FOR LEG NO.
L
95
9.1
Mine
Firing
Routine
TREF-DT
<0
No
86
Yes
96
9.2
Submarine
Detection-
Attrition
Routine
101

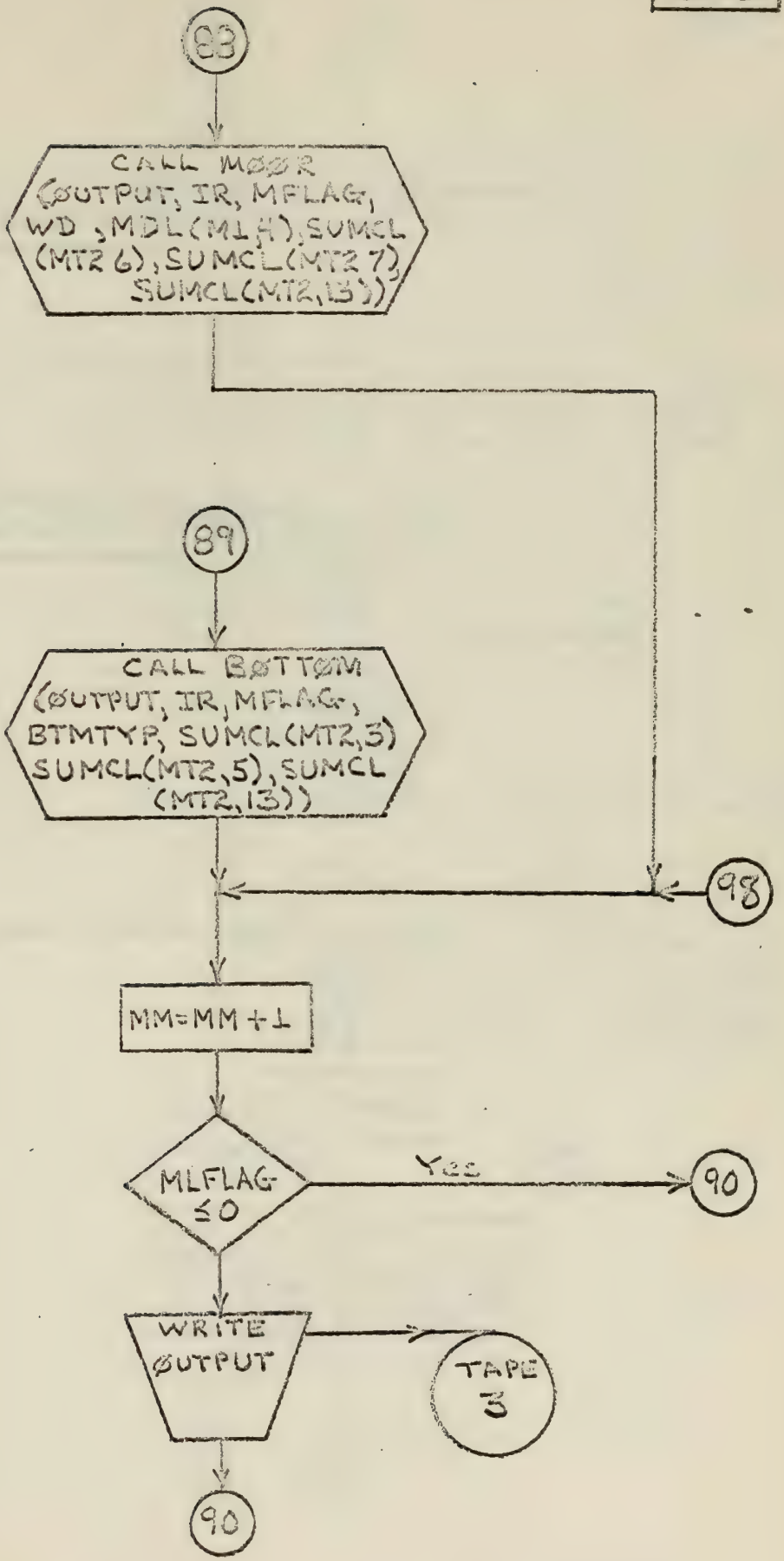
9. CONTD


86
TREF-TDIF <0
No
Yes
PRINT TAM, SUB NØ., VAL(I,3), ABØRT MISSION DUE TØ TIME LIMITATION. MINES REMAINING ØN BOARD ARE
97
9.3
Mine Posit and Depth Routine
9.4
Mines Remaining Routine
101
WD≤0
No
87
Yes
ØUTPUT(9)=1.
CALL TAPØ (ØUTPUT, TLAY, MT2, MDL(M1,3), MDL(M1,4), MDL(M1,7), MDL(M1,5), MDL(M1,6), M1, XPØSIT, YPØSIT, WD, MFLAG)
98

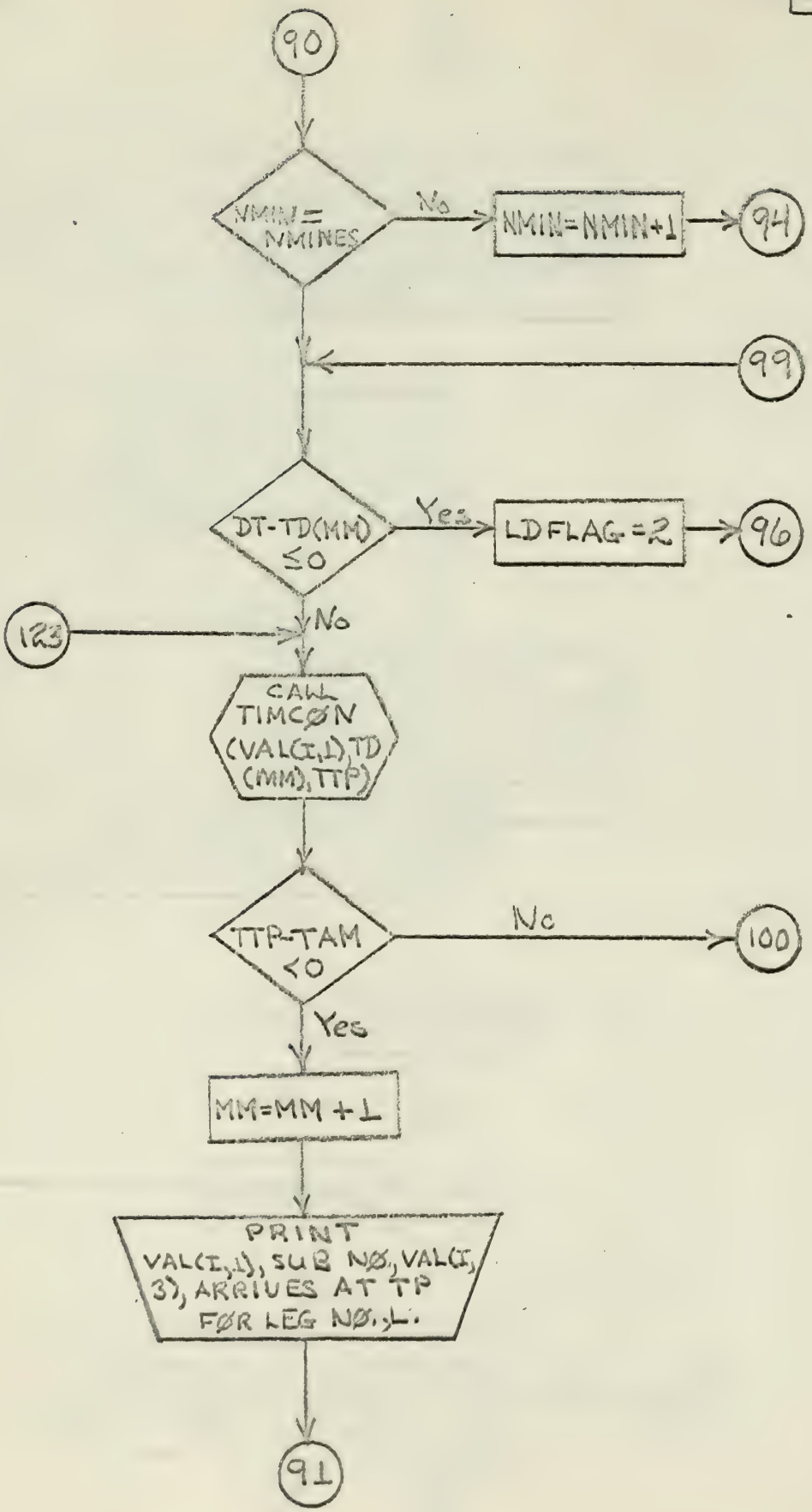
9. CONTD


87
CALL TAPØ (ØUTPUT, TLAY, MTR, MDL(ML,3), MDL(ML,8), MDL(ML,7), MDL(ML,5), MDL(ML,6), ML, XPØSIT, YPØSIT, WD, MFLAG)
CALL DEPTH (XPØSIT, YPØSIT, BØT, NMAXL, MMAXL, YMEAN, XMEANX, BTMTYP)
CALL RAN1 (IR, RN)
PD = SUMCL(MTR,2) / 5. * BTMTYP
PD-RN <0
Yes
No
ØUTPUT(3)=1.
GØ TO MFLAG
MFLAG=3,4
89
MFLAG=1,2,5,6
88

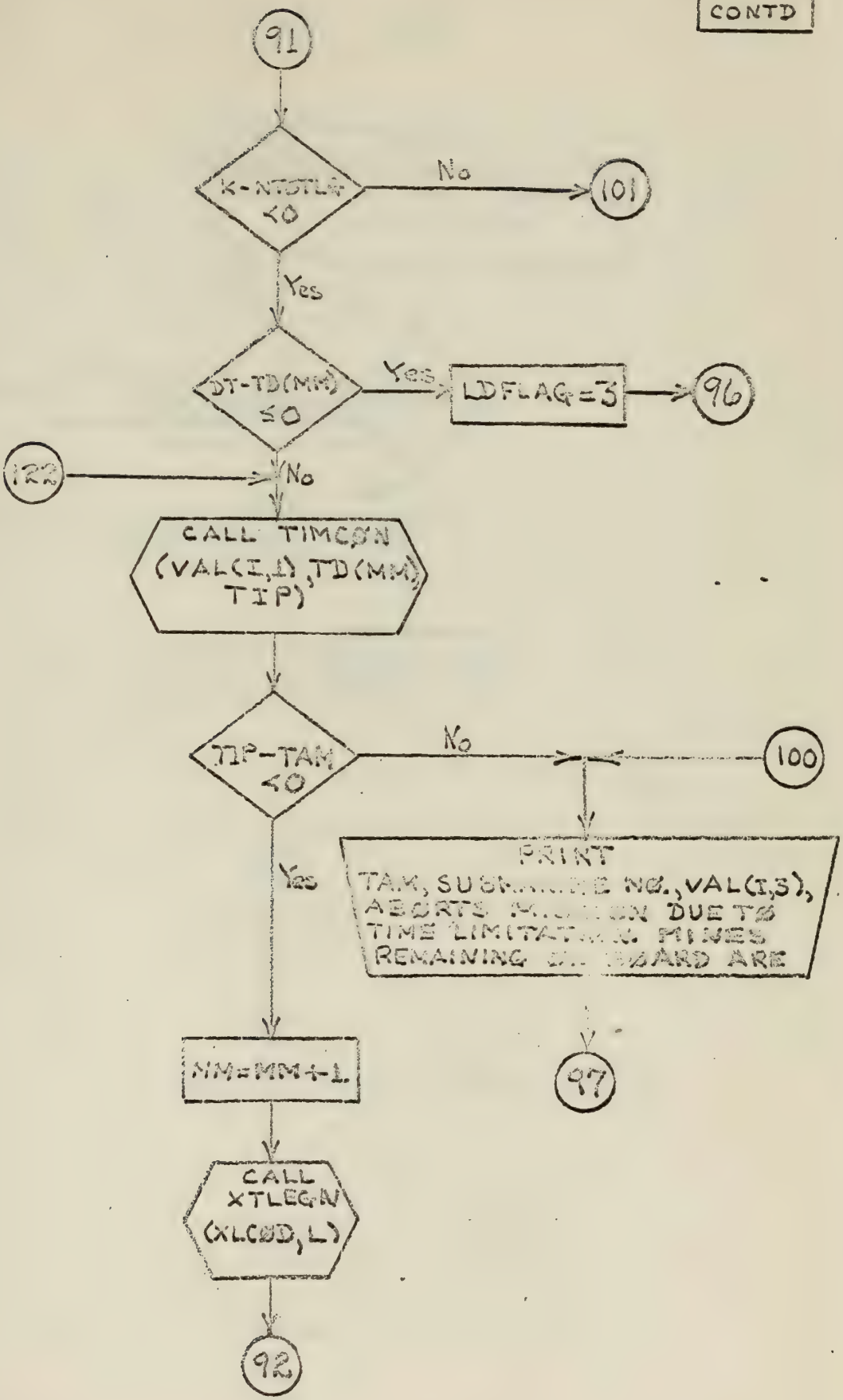
9. CONTD

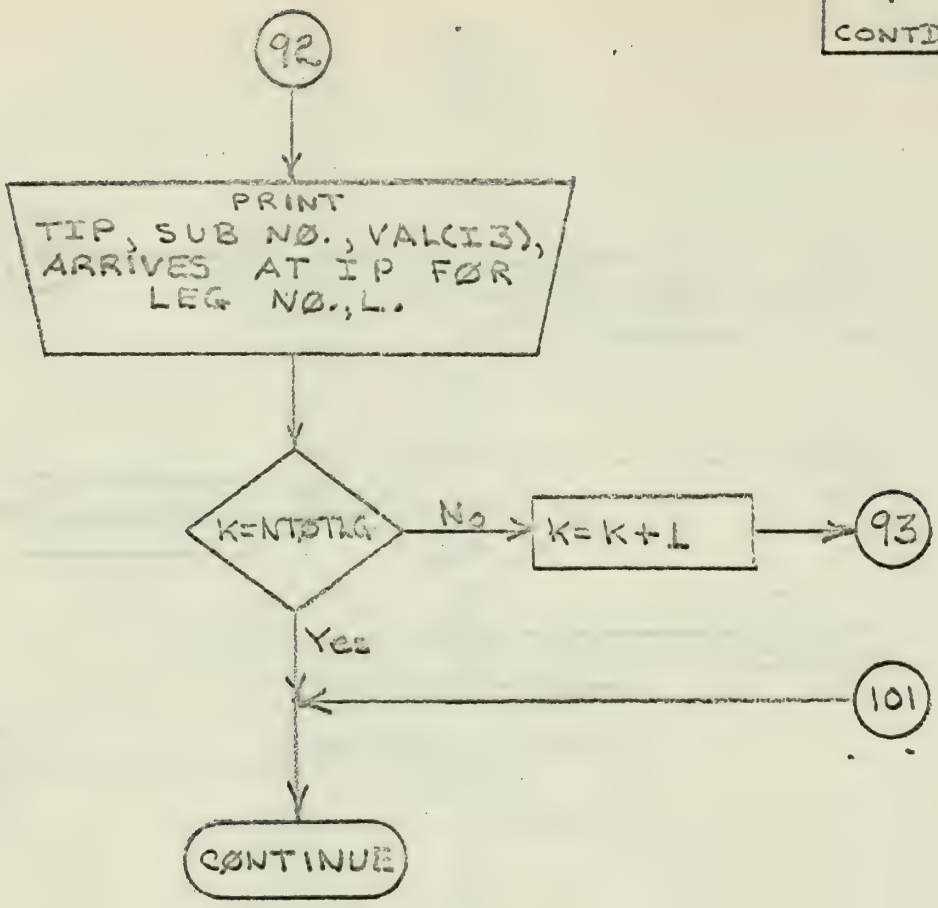

88
CALL MØØR (ØUTPUT, IR, MFLAG, WD, MDL(M1,H), SUMCL (MTR 6), SUMCL(MTR 7), SUMCL(MTR,13))
89
CALL BØTTØM (ØUTPUT, IR, MFLAG, BTMTYP, SUMCL(MTR,3) SUMCL(MTR,5), SUMCL (MTR,13))
98
MM=MM+1
MLFLAG ≤0
Yes
90
WRITE ØUTPUT
TAPE 3
90

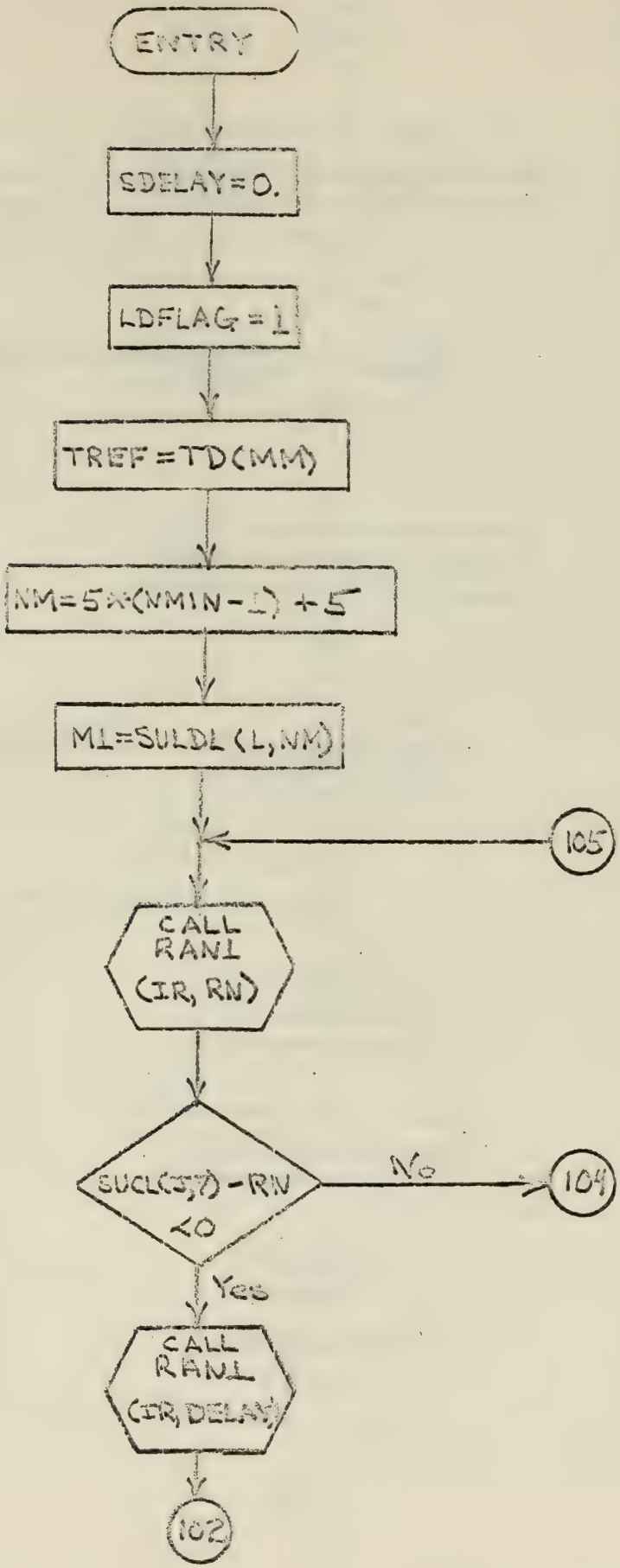
9.
CONTD


90
NMIN = NMINES
No
NMIN=NMIN+1
94
99
DT-TD(MM) ≤0
Yes
LDFLAG=2
96
123
No
CALL TIMCØN (VAL(I,1),TD (MM),TTP)
TTP-TAM <0
No
100
Yes
MM=MM+1
PRINT VAL(I,1), SUB NØ, VAL(I, 3), ARRIVES AT TP FØR LEG NØ, L.
91

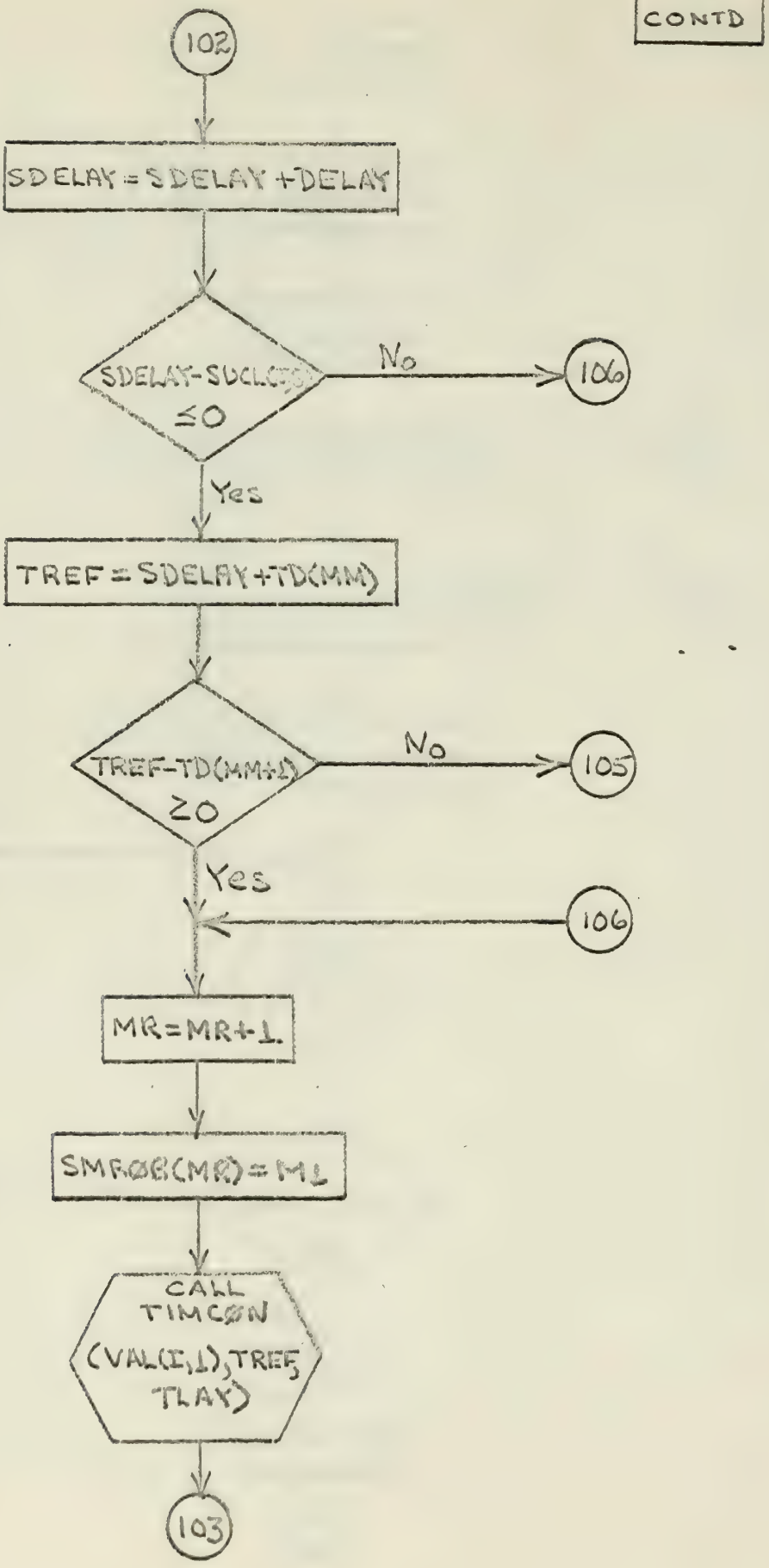
9. CONTD


91
K-NTOTL≤ <0
No
101
Yes
DT-TD(MM) ≤0
Yes
LDFLAG=3
96
122
No
CALL TIMCON (VAL(I,1),TD(MM), TIP)
TIP-TAM <0
No
100
Yes
PRINT
TAM, SU... NO., VAL(I,3),
ABORTS M... ON DUE TO
TIME LIMITATI... MINES
REMAINING ... BOARD ARE
97
MM=MM+1
CALL XTLEG (XLCOD, L)
92

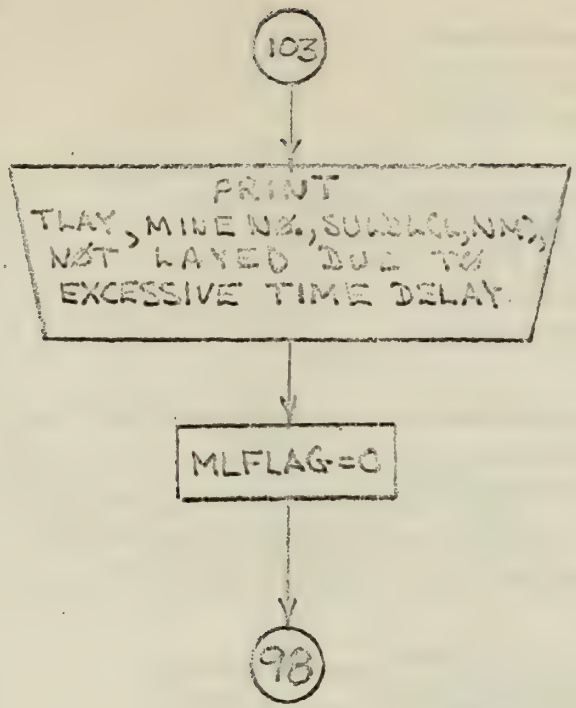

92
PRINT
TIP, SUB NØ., VAL(I3),
ARRIVES AT IP FØR
LEG NØ., L.
K=NTØTLG
No
K=K+L
93
Yes
101
CØNTINUE

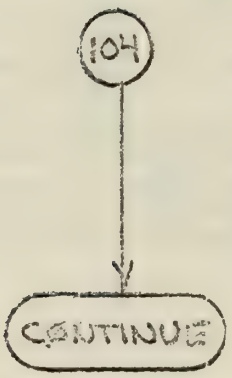

ENTRY
SDELAY=0.
LDFLAG=1
TREF=TD(MM)
NM=5*(NMIN-1)+5
M1=SULDL(L,NM)
105
CALL RANI (IR, RN)
SUCL(J,Y)-RN <0
No
104
Yes
CALL RANI (IR, DELAY)
102

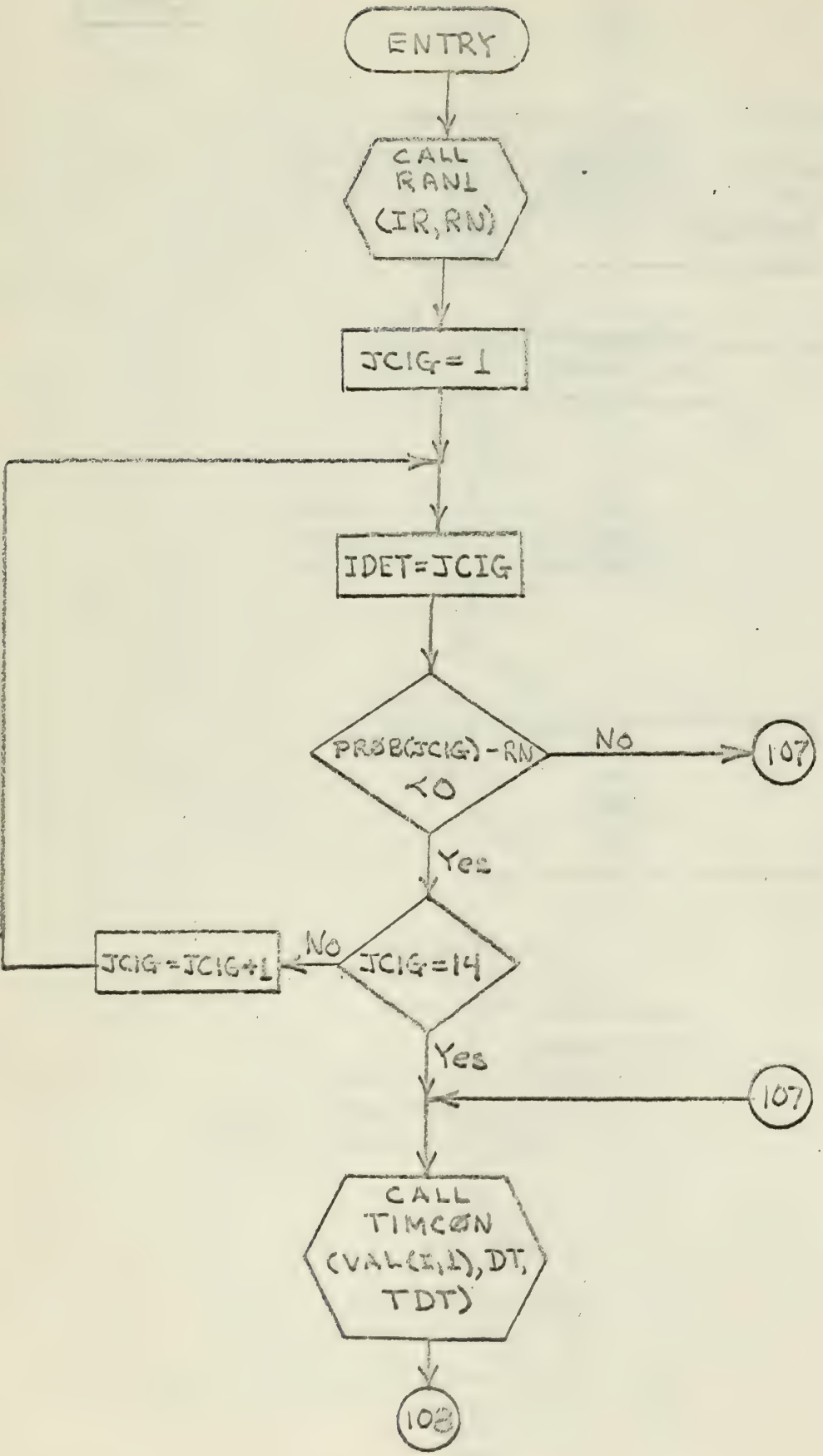
9.1 CONTD


102
SDELAY = SDELAY + DELAY
SDELAY-SUCLCT(3)
≤0
No
106
Yes
TREF = SDELAY + TD(MM)
TREF-TD(MM+1)
≥0
No
105
Yes
106
MR = MR+1.
SMROB(MR) = ML
CALL TIMCON (VAL(I,1), TREF, TLAY)
103

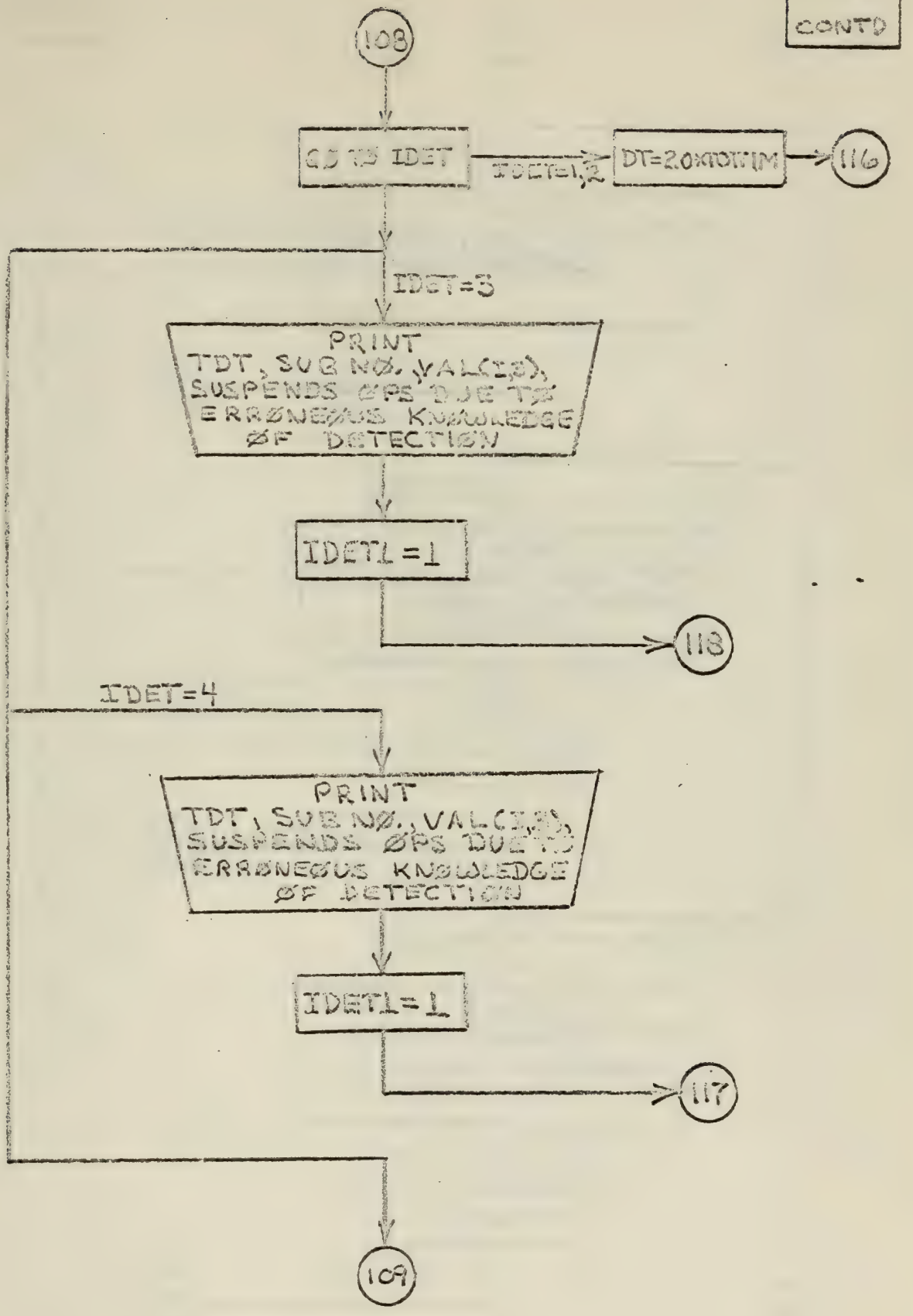
9.1
CONTD


103
PRINT
TLAY, MINE NO., SUBFLAG, NM,
NOT LAYED DUE TO
EXCESSIVE TIME DELAY.
MLFLAG=0
98

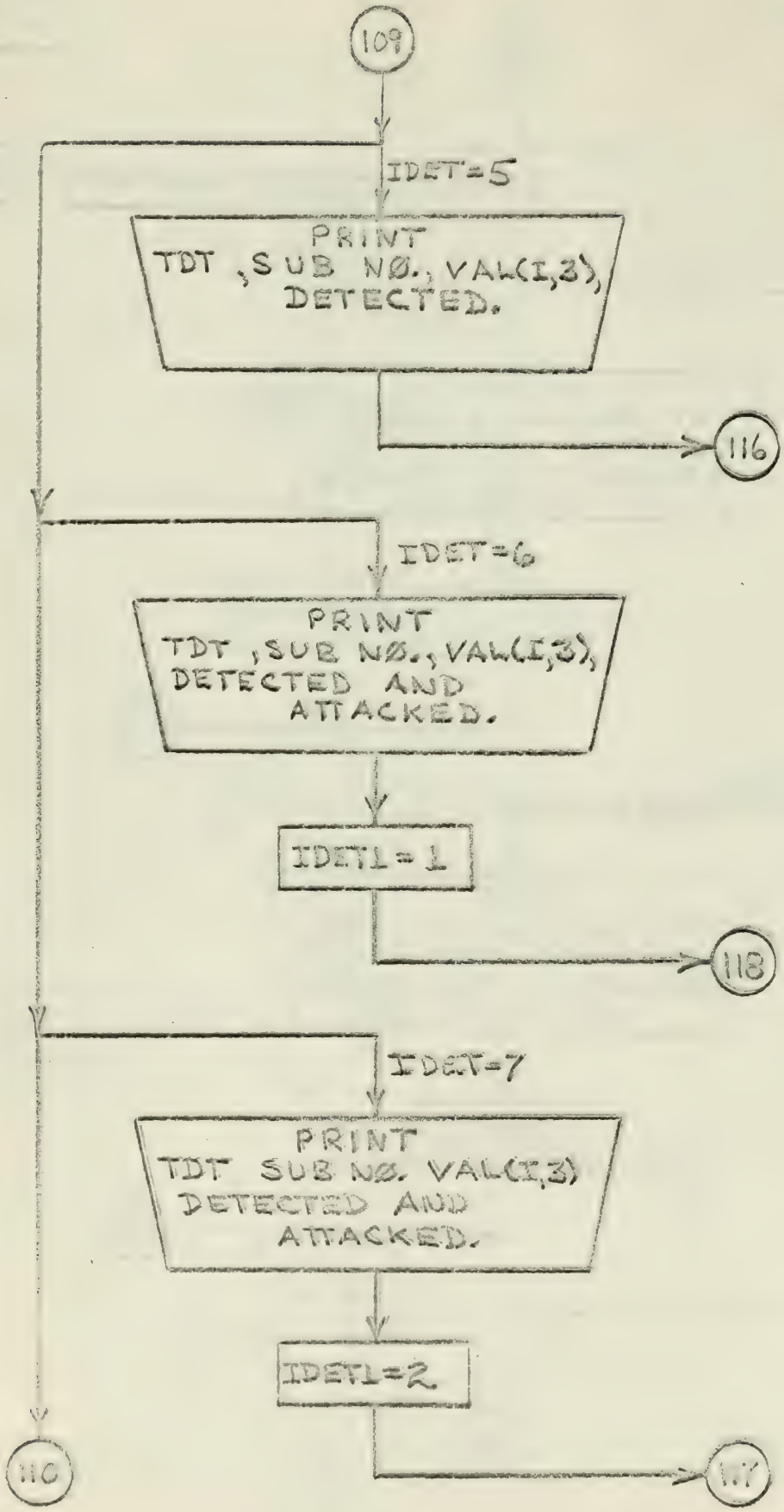

104
CONTINUE


ENTRY
CALL RANL (IR,RN)
JCIG=1
IDET=JCIG
PROB(JCIG)-RN <0
No
107
Yes
JCIG=JCIG+1
No
JCIG=14
Yes
107
CALL TIMCON (VAL(I,1),DT, TDT)
108

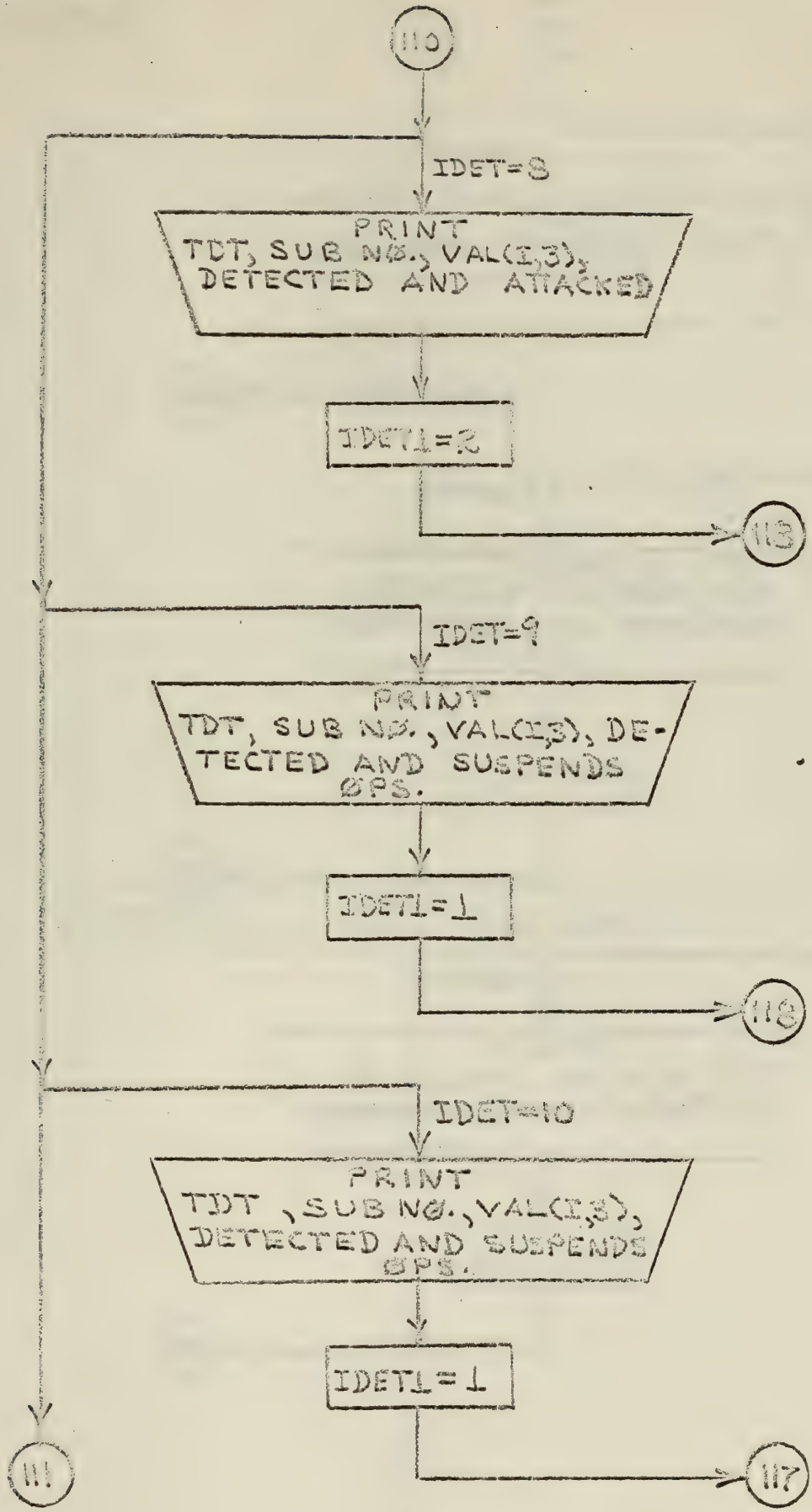
9.2 CONTD


108
GO TO IDET
IDET=1,2
DT=2.0*10**-7*M
116
IDET=3
PRINT TDT, SUB NØ., VAL(I,3), SUSPENDS ØPS DUE TØ ERRØNEØUS KNØWLEDGE ØF DETECTION
IDETL=1
118
IDET=4
PRINT TDT, SUB NØ., VAL(I,3), SUSPENDS ØPS DUE TØ ERRØNEØUS KNØWLEDGE ØF DETECTION
IDETL=1
117
109

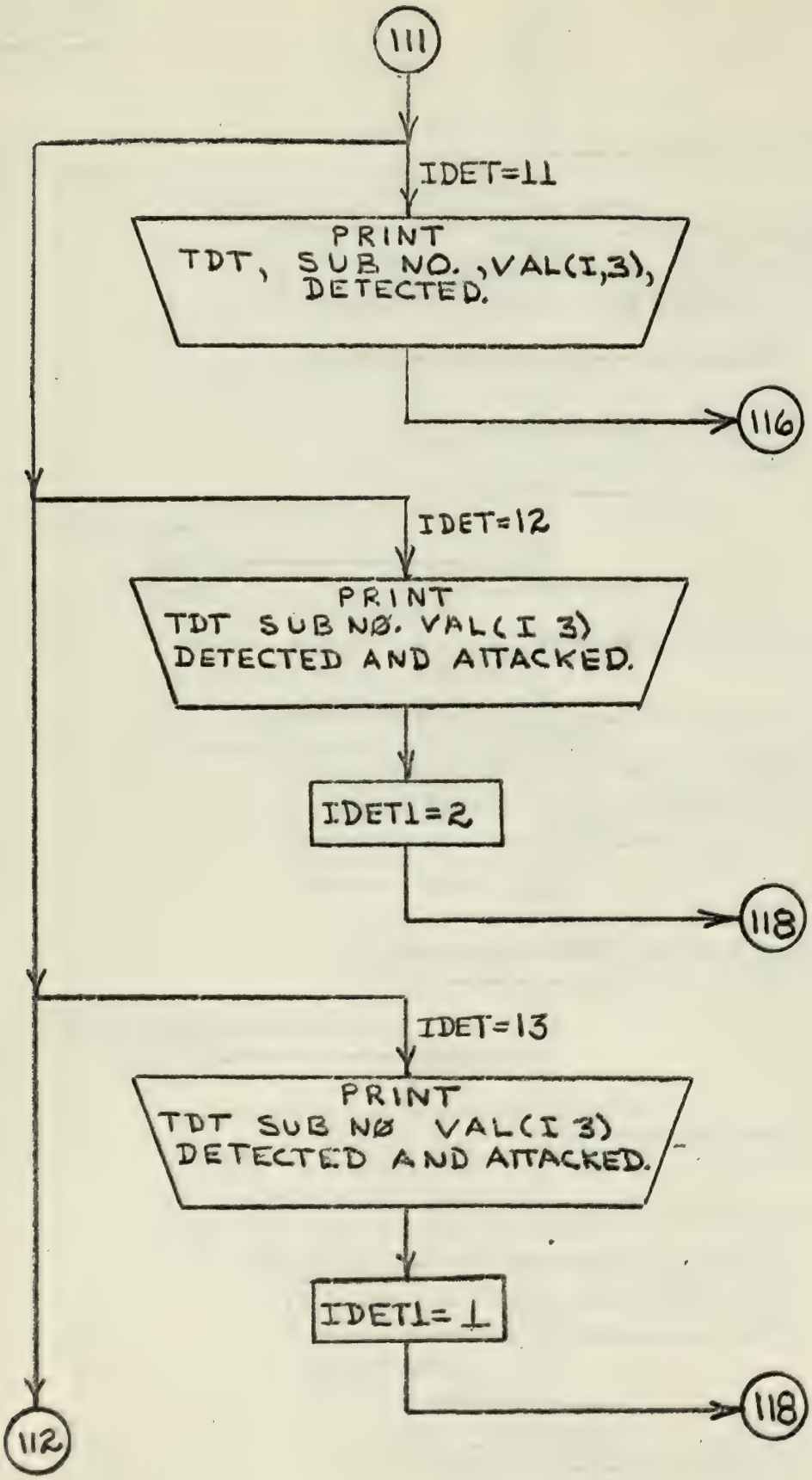
9.2 CONTD


109
IDET=5
PRINT
TDT ,SUB NØ., VAL(I,3),
DETECTED.
116
IDET=6
PRINT
TDT ,SUB NØ., VAL(I,3),
DETECTED AND
ATTACKED.
IDETL=1
118
IDET=7
PRINT
TDT SUB NØ. VAL(I,3)
DETECTED AND
ATTACKED.
IDETL=2.
110
117

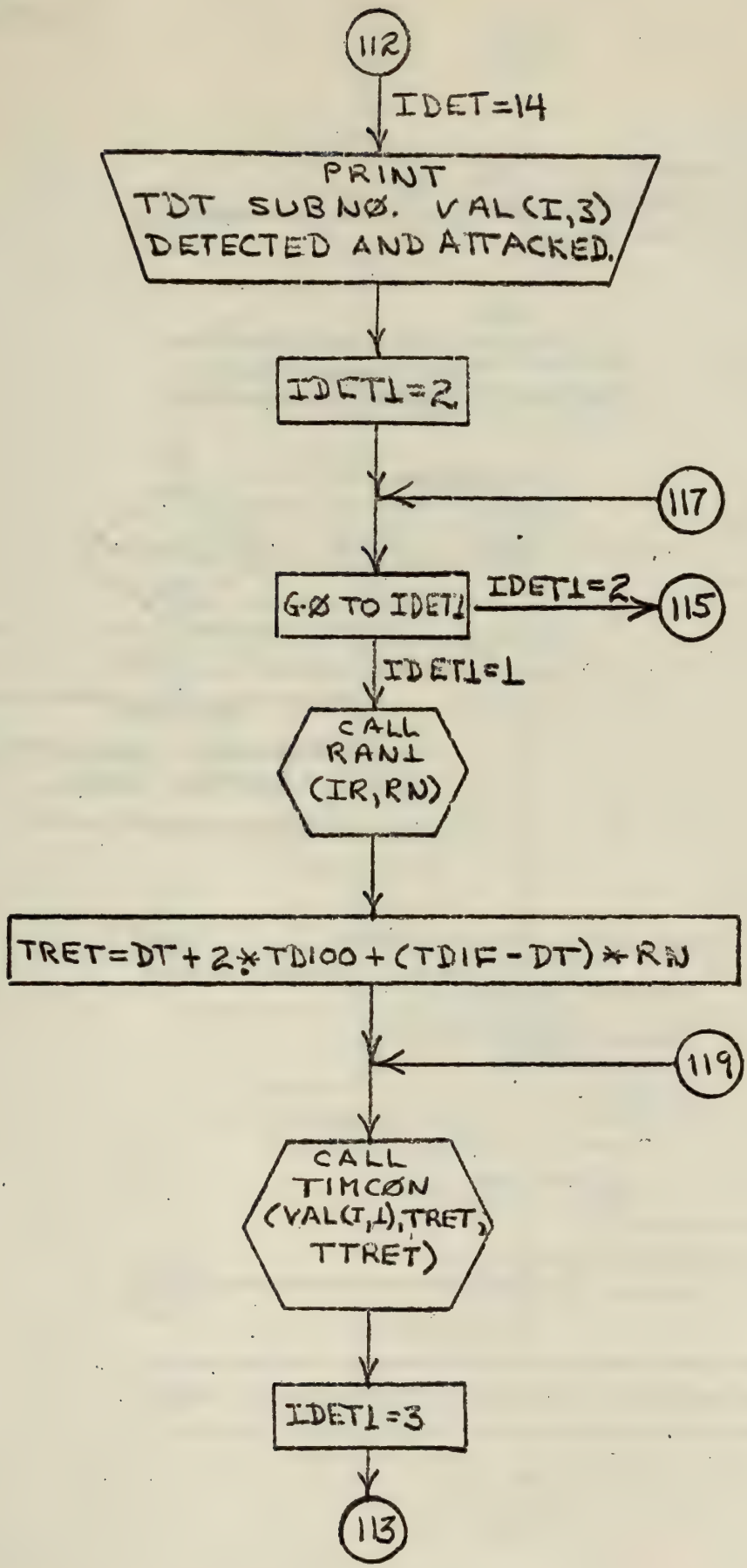
9.2 CONTD


110
IDET=8
PRINT TDT, SUB NØ., VAL(I,3), DETECTED AND ATTACKED
IDETL=2
113
IDET=9
PRINT TDT, SUB NØ., VAL(I,3), DETECTED AND SUSPENDS ØPS.
IDETL=1
118
IDET=10
PRINT TDT, SUB NØ., VAL(I,3), DETECTED AND SUSPENDS ØPS.
IDETL=1
111
117

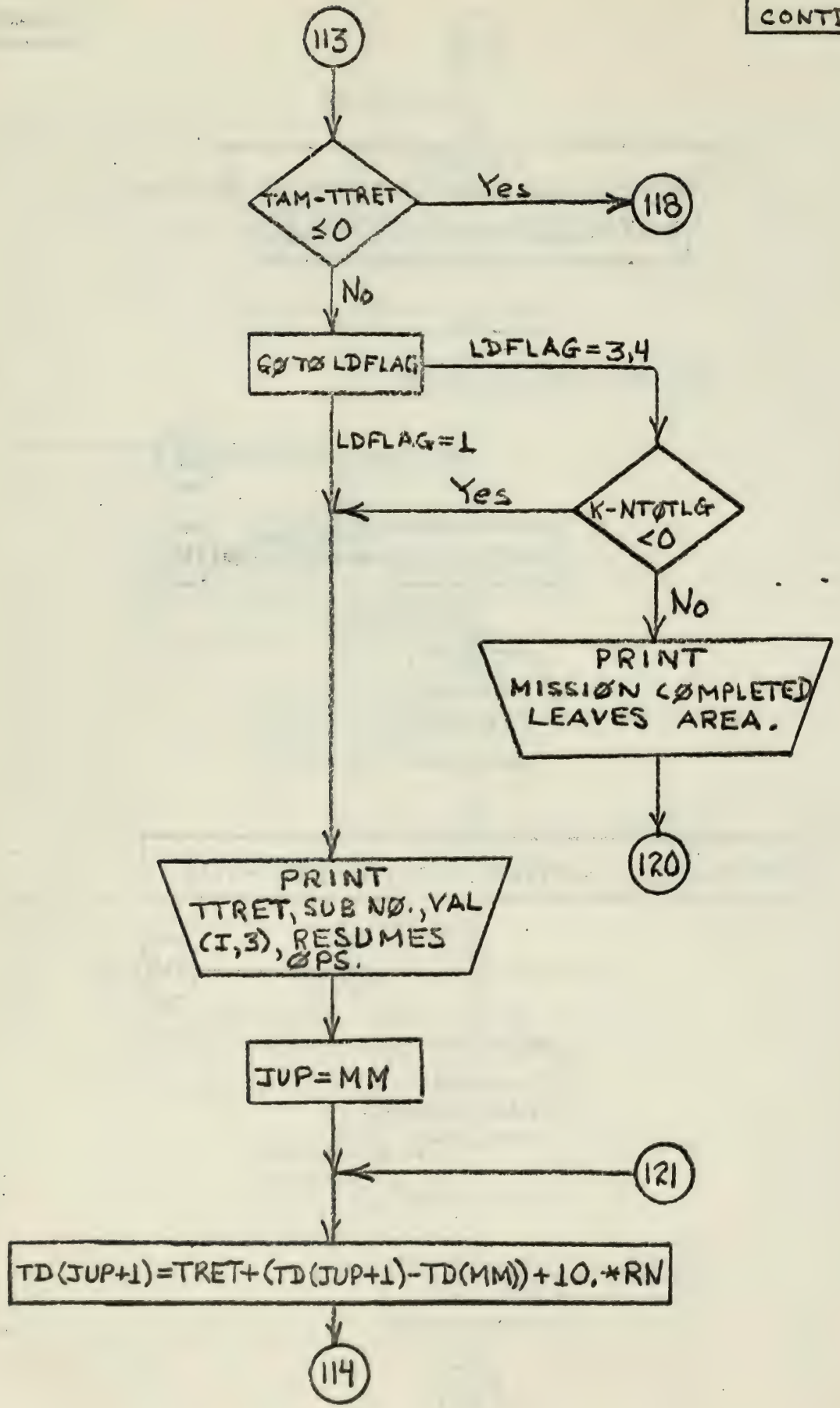
9.2 CONTD


111
IDET=11
PRINT
TDT, SUB NO. ,VAL(I,3),
DETECTED.
116
IDET=12
PRINT
TDT SUB NØ. VAL(I 3)
DETECTED AND ATTACKED.
IDET1=2
118
IDET=13
PRINT
TDT SUB NØ VAL(I 3)
DETECTED AND ATTACKED.
IDET1=1
118
112

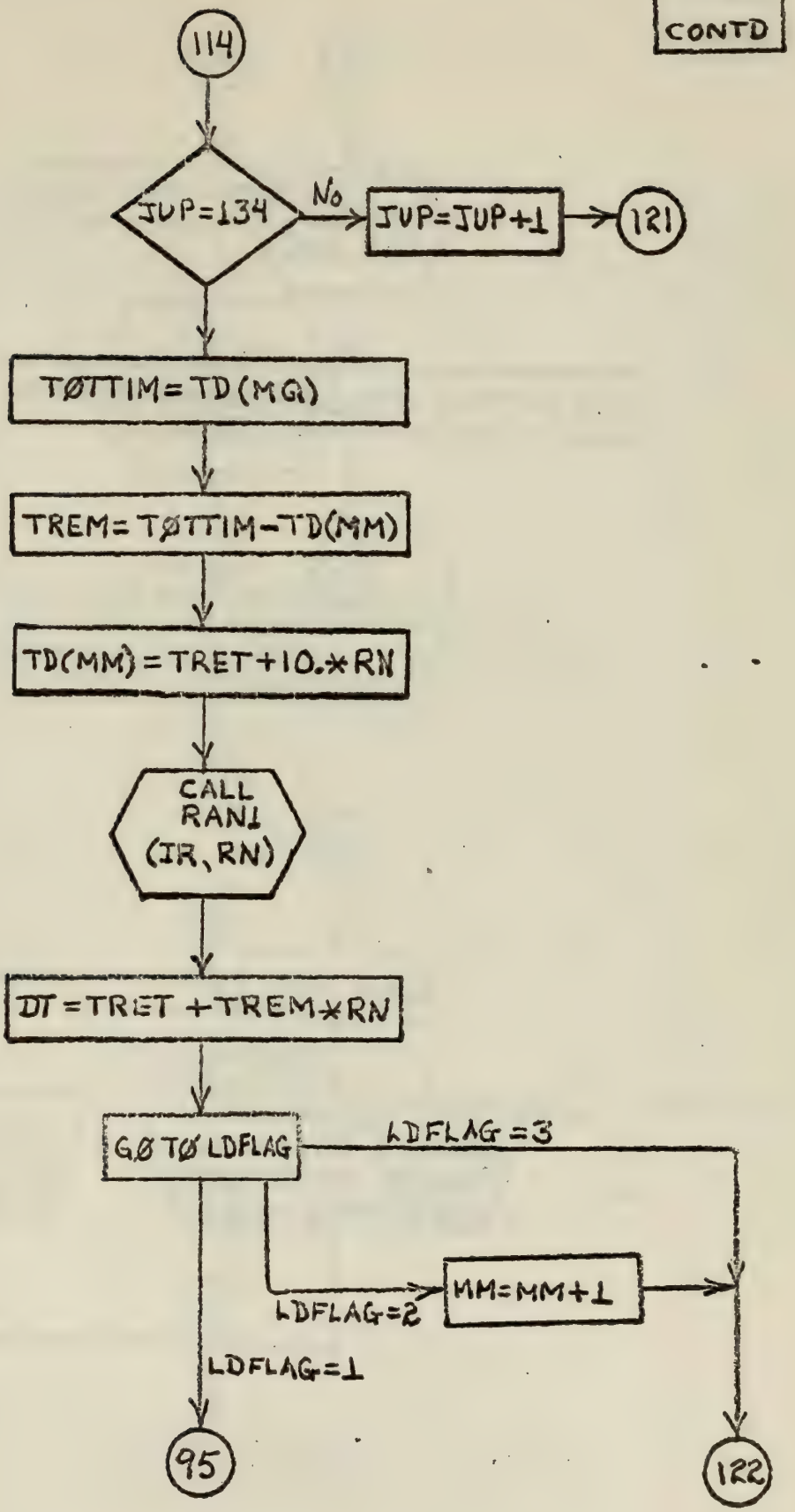
9.2 CONTD


112
IDET=14
PRINT
TDT SUBNØ. VAL(I,3)
DETECTED AND ATTACKED.
IDET1=2
117
G-Ø TO IDET1
IDET1=2
115
IDET1=1
CALL
RAN1
(IR,RN)
TRET=DT+2*TD100+(TDIF-DT)*RN
119
CALL
TIMCØN
(VAL(I,1),TRET,
TTRET)
IDET1=3
113

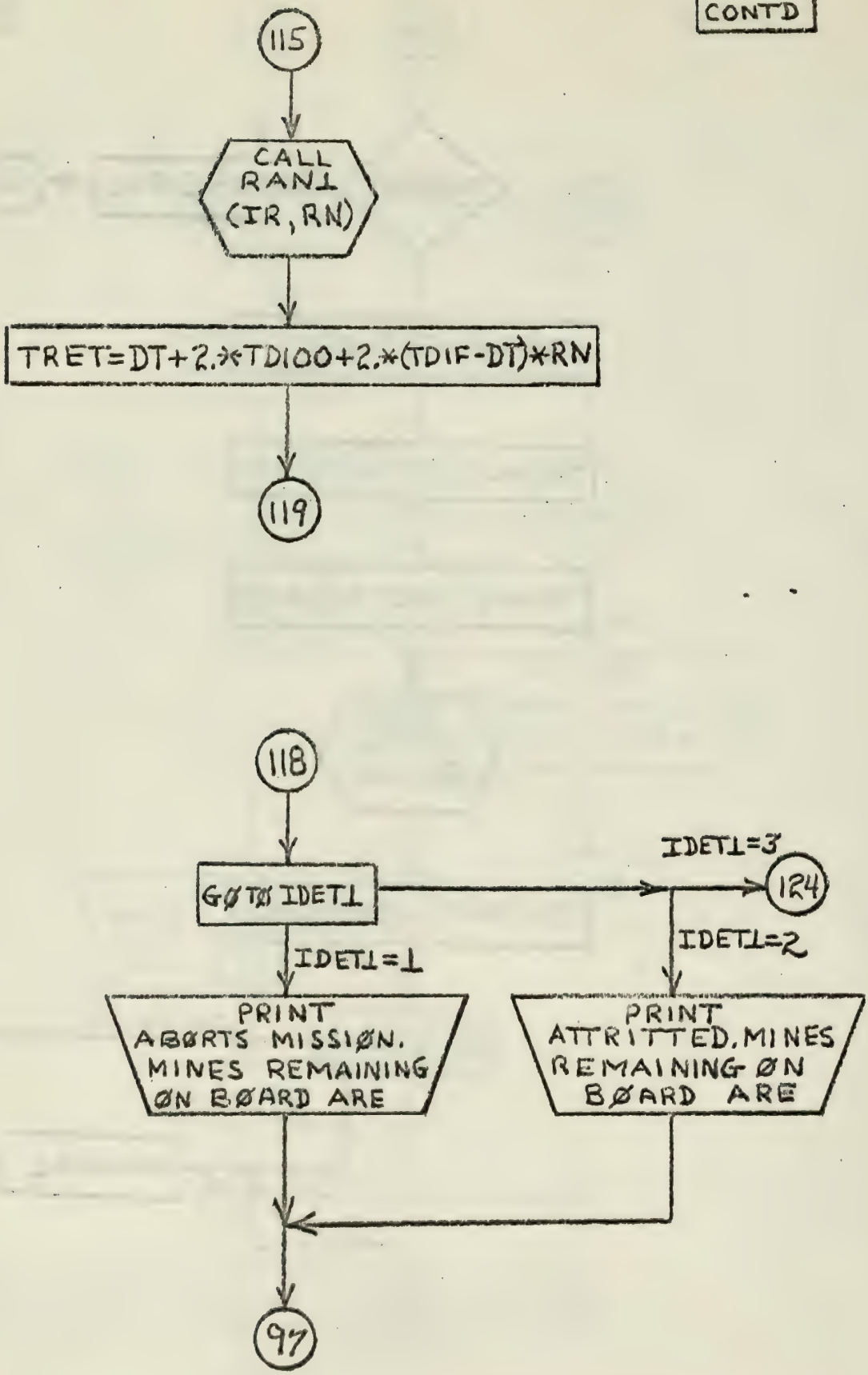
9.2 CONTD


113
TAM-TTRET ≤0
Yes
118
No
GØTØ LDFLAG
LDFLAG=3,4
LDFLAG=1
Yes
K-NTØTLG <0
No
PRINT MISSIØN CØMPLETED LEAVES AREA.
120
PRINT TTRET, SUB NØ., VAL (I,3), RESUMES ØPS.
JUP=MM
121
TD(JUP+1)=TRET+(TD(JUP+1)-TD(MM))+10.*RN
114

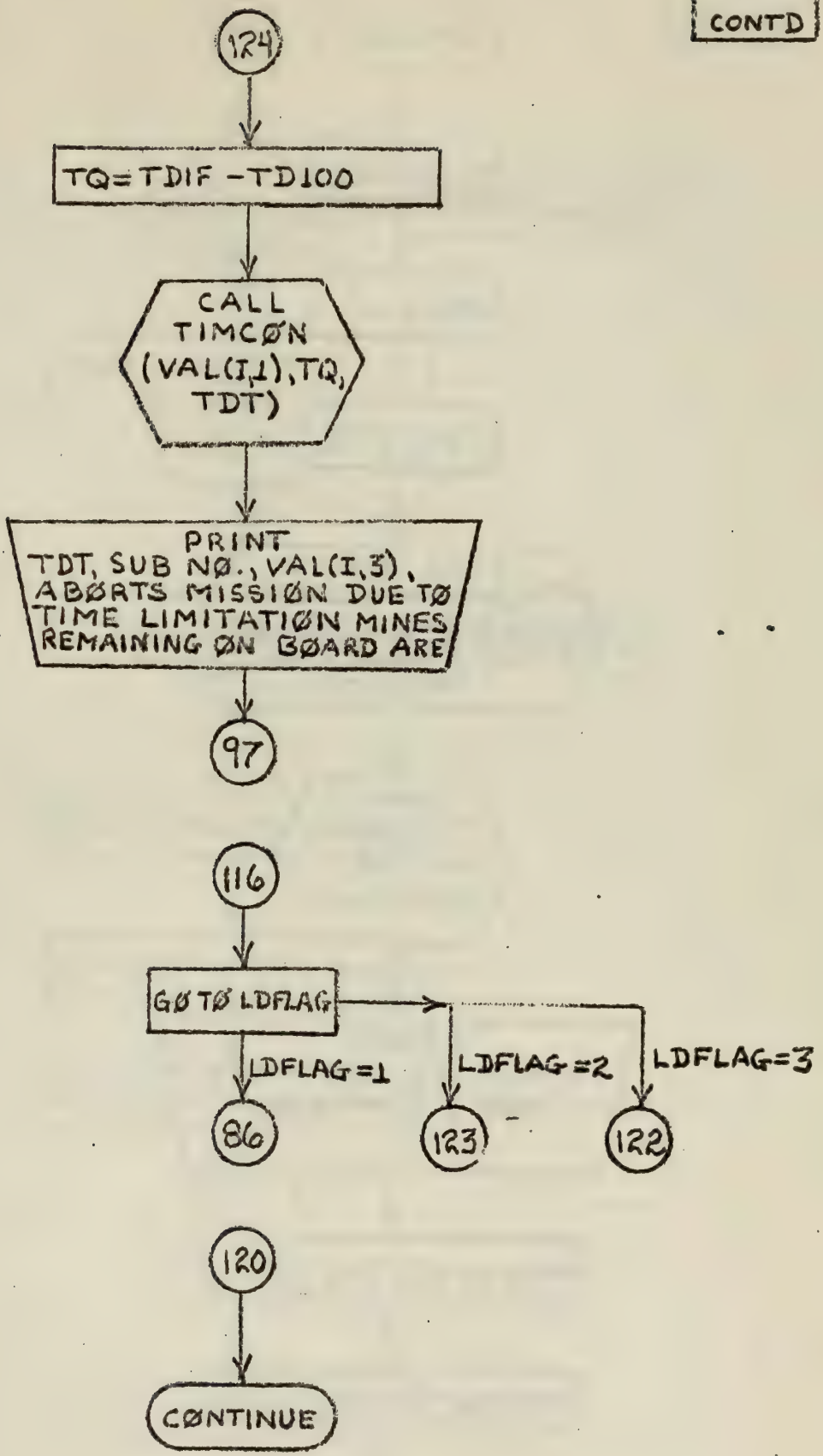
9.2 CONTD

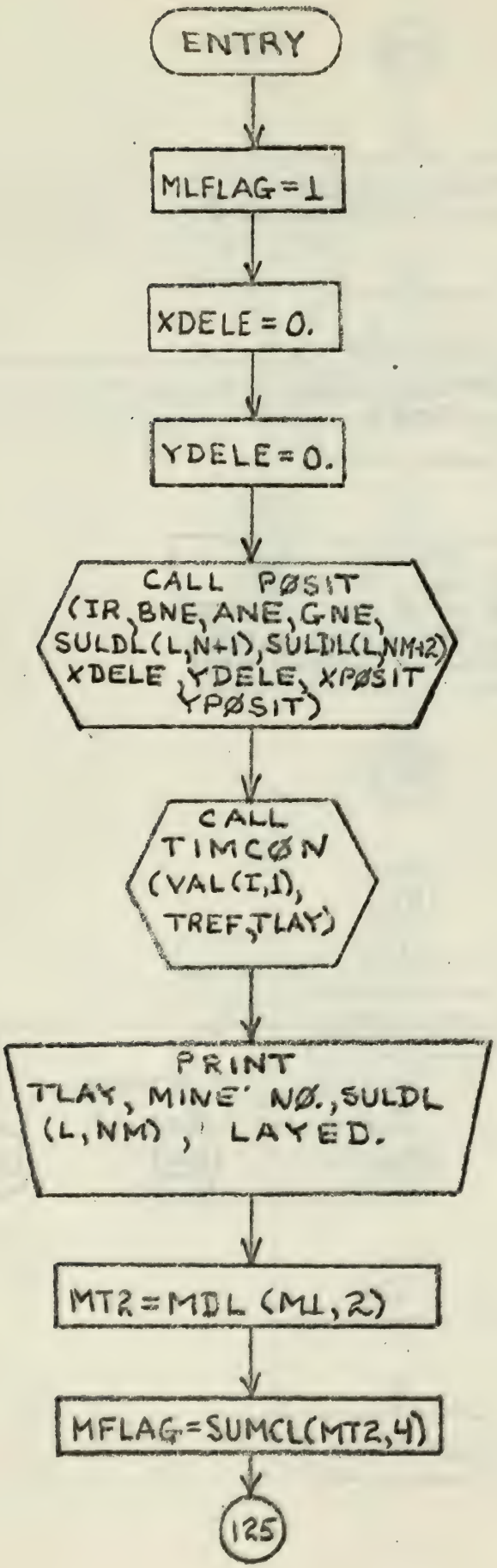

114
JUP=134
No
JUP=JUP+1
121
TØTTIM=TD(MG)
TREM=TØTTIM-TD(MM)
TD(MM)=TRET+10.*RN
CALL RANI (IR, RN)
DT=TRET+TREM*RN
GØ TØ LDFLAG
LDFLAG=3
LDFLAG=2
MM=MM+1
LDFLAG=1
95
122

9.2 CONTD

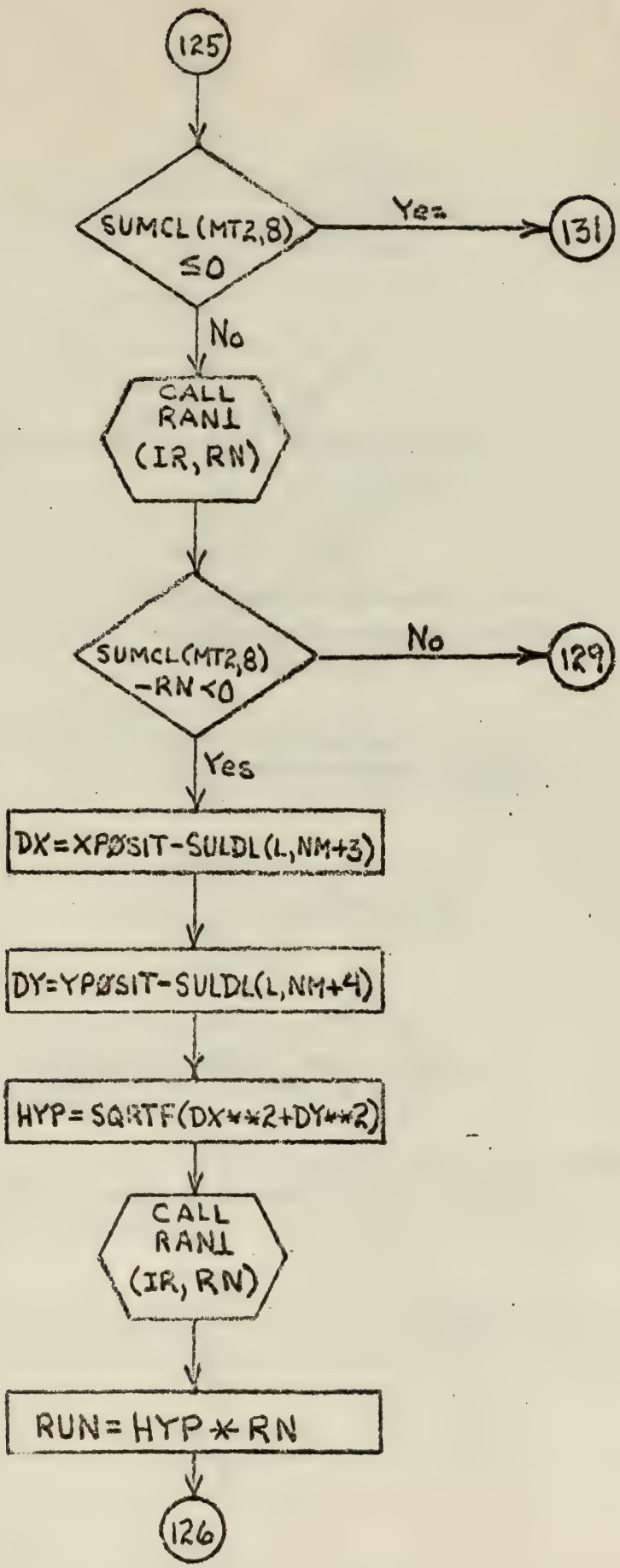

115
CALL RANU (IR, RN)
TRET=DT+2.*TD100+2.*(TDIF-DT)*RN
119
118
GØTØ IDETU
IDETU=3
124
IDETU=1
IDETU=2
PRINT ABØRTS MISSIØN. MINES REMAINING ØN BØARD ARE
PRINT ATTRITTED. MINES REMAINING ØN BØARD ARE
97

9.2
CONTD

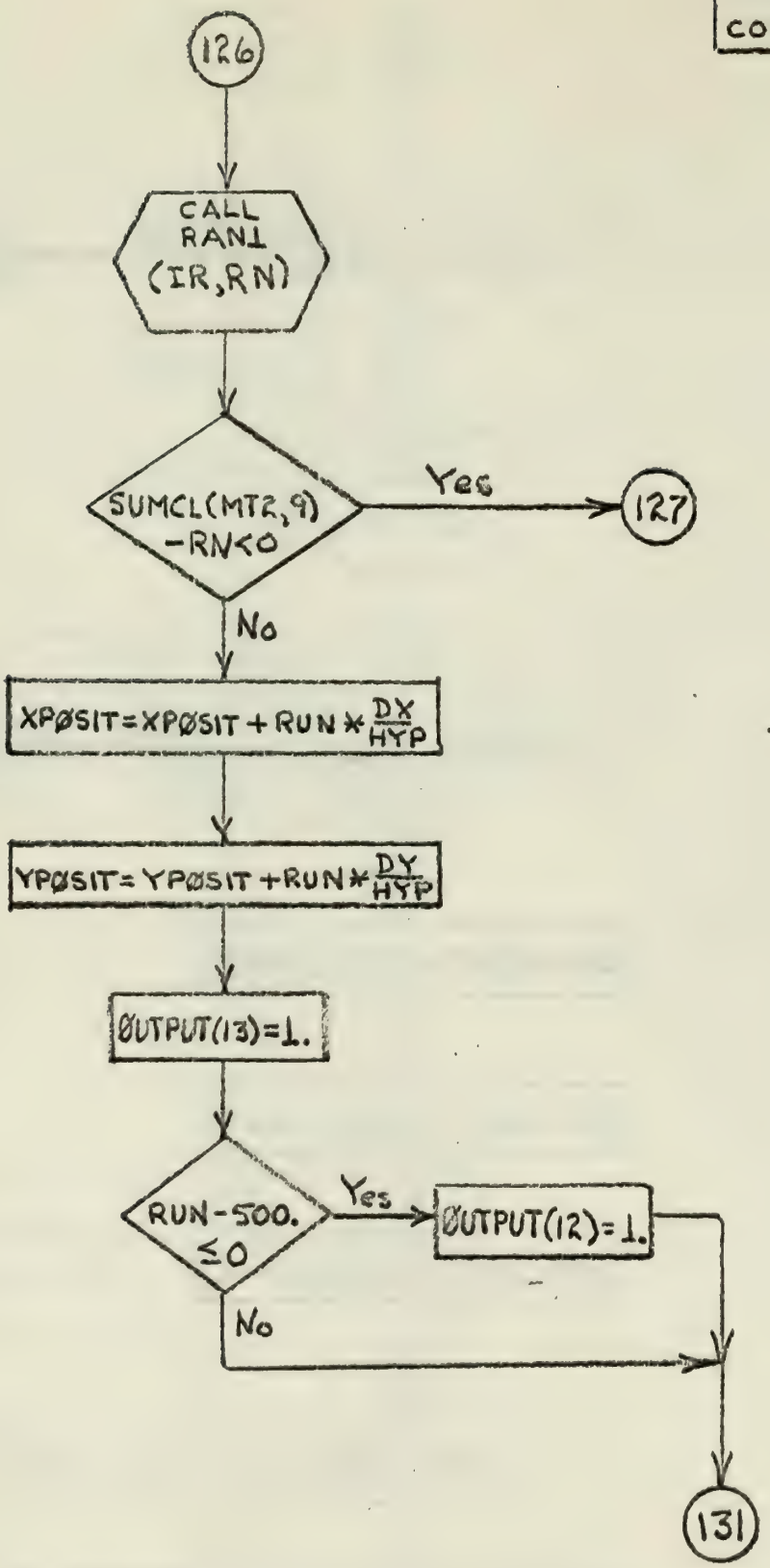

124
TQ=TDIF -TD100
CALL TIMCØN (VAL(I,1),TQ, TDT)
PRINT TDT, SUB NØ., VAL(I,3), ABØRTS MISSIØN DUE TØ TIME LIMITATIØN MINES REMAINING ØN BØARD ARE
97
116
GØ TØ LDFLAG
LDFLAG=1
LDFLAG=2
LDFLAG=3
86
123
122
120
CØNTINUE


ENTRY
MLFLAG=1
XDELE=0.
YDELE=0.
CALL PØSIT (IR, BNE, ANE, GNE, SULDL(L,N+1), SULDL(L,NM+2), XDELE, YDELE, XPØSIT, YPØSIT)
CALL TIMCØN (VAL(I,1), TREF, TLAY)
PRINT TLAY, MINE' NØ., SULDL (L,NM), ' LAYED.
MT2=MDL (M1,2)
MFLAG=SUMCL(MT2,4)
125

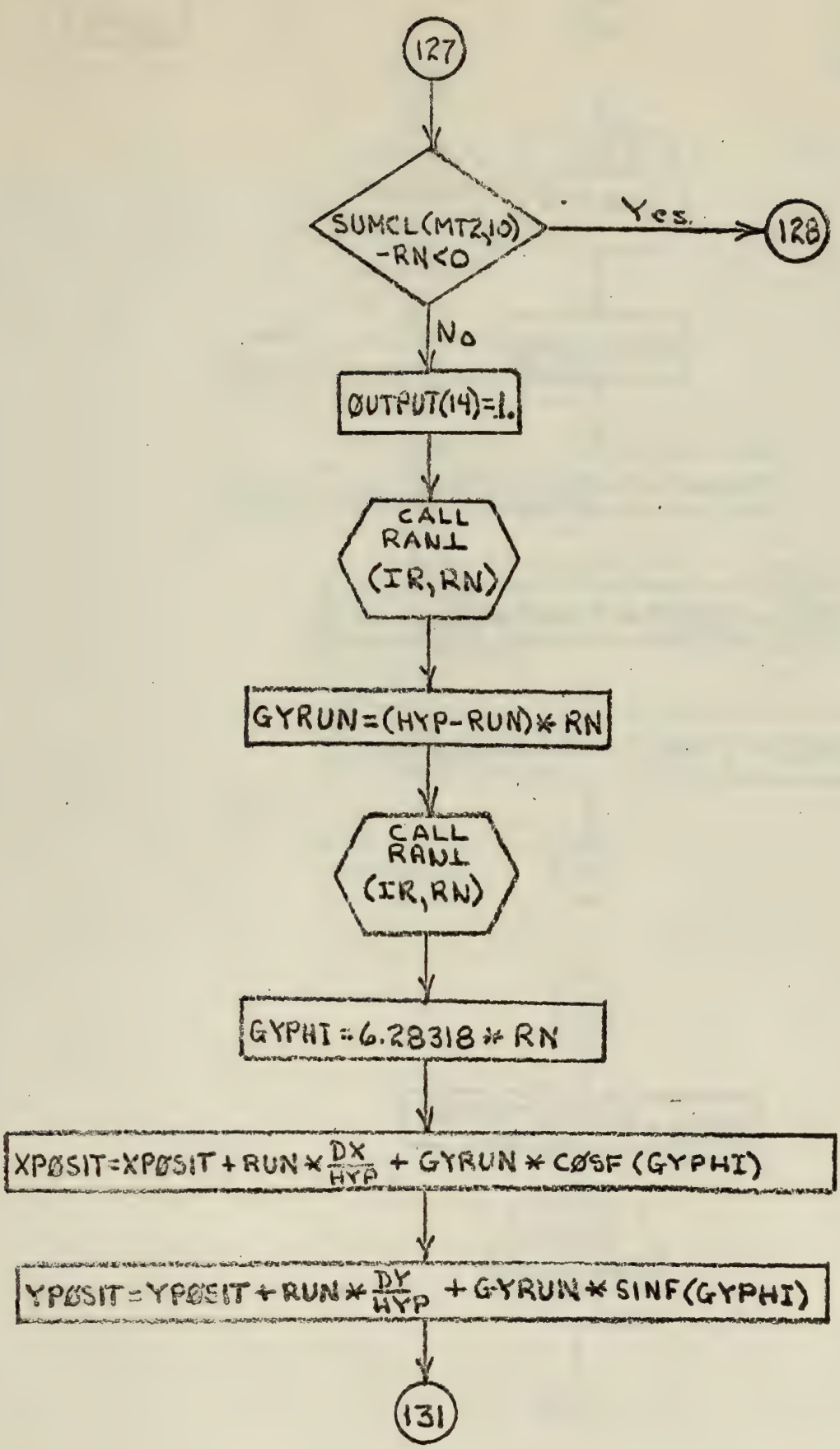
9.3 CONTD


125
SUMCL(MT2,8) ≤0
Yes
131
No
CALL RANI (IR,RN)
SUMCL(MT2,8) -RN<0
No
129
Yes
DX=XPØSIT-SULDL(L,NM+3)
DY=YPØSIT-SULDL(L,NM+4)
HYP=SQRTF(DX**2+DY**2)
CALL RANI (IR,RN)
RUN=HYP*RN
126

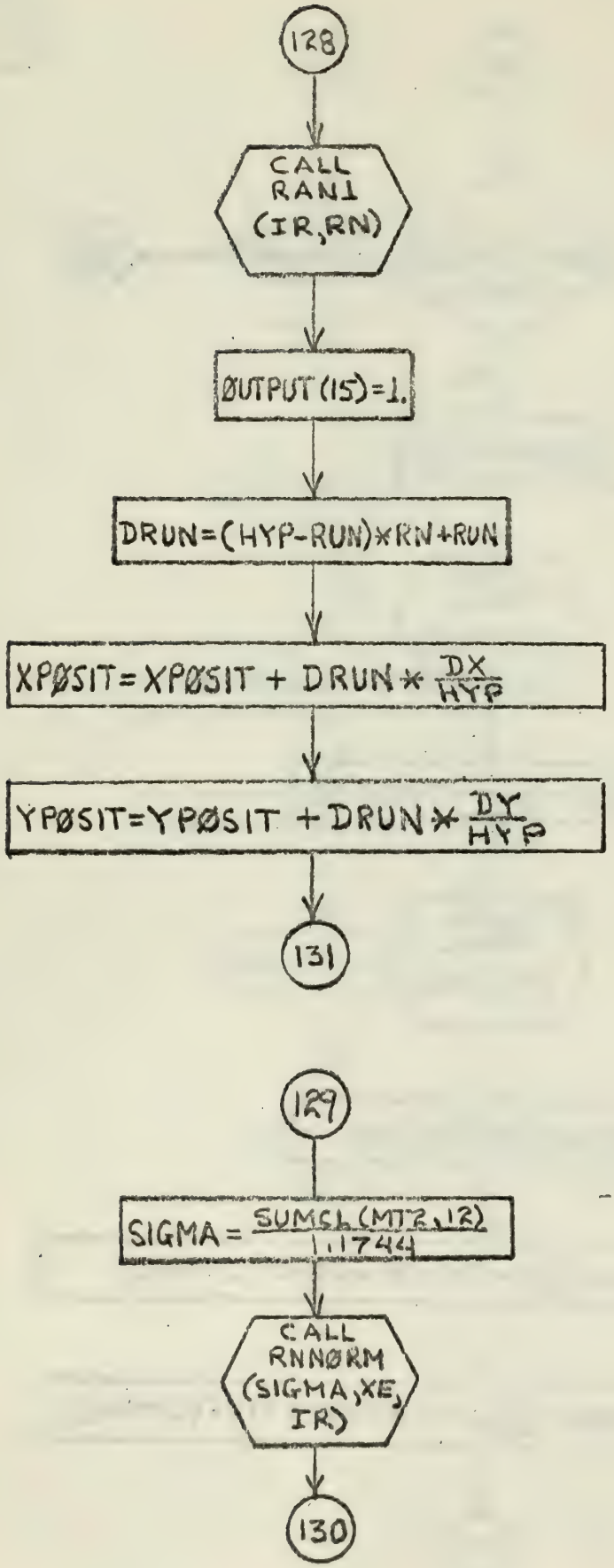
9.3 CONTD


126
CALL RANI (IR,RN)
SUMCL(MT2,9) -RN<0
Yes
127
No
XPØSIT=XPØSIT + RUN * DX/HYP
YPØSIT=YPØSIT +RUN * DY/HYP
ØUTPUT(13)=1.
RUN-500. ≤0
Yes
ØUTPUT(12)=1.
No
131

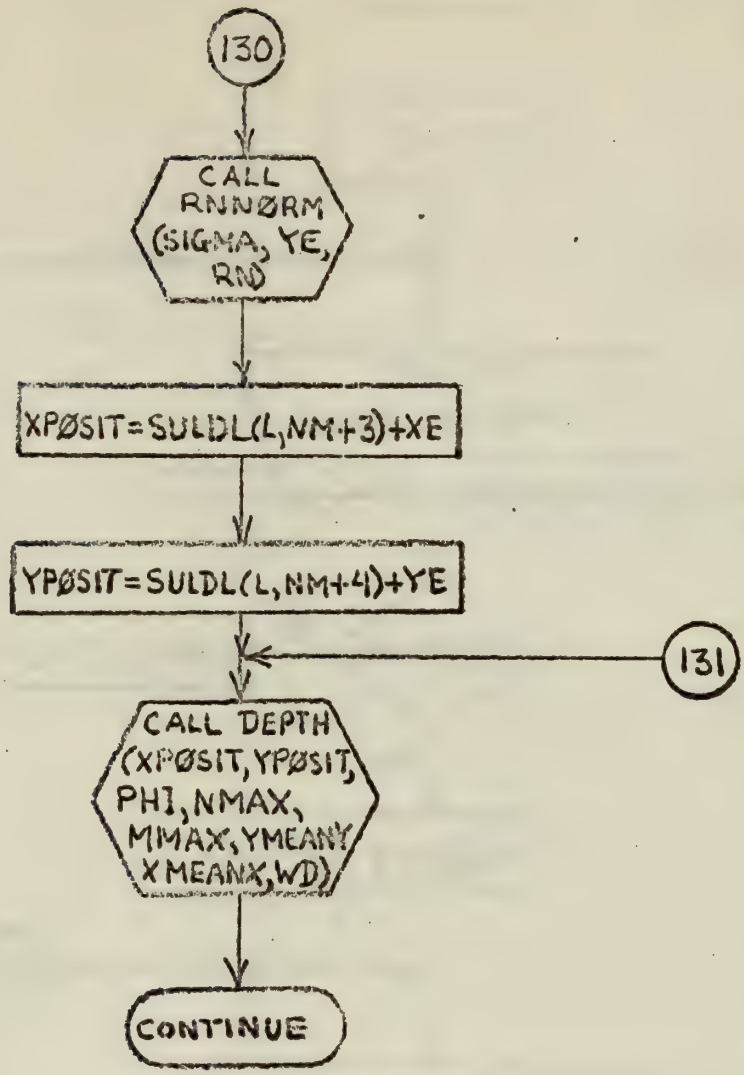
9.3 CONTD

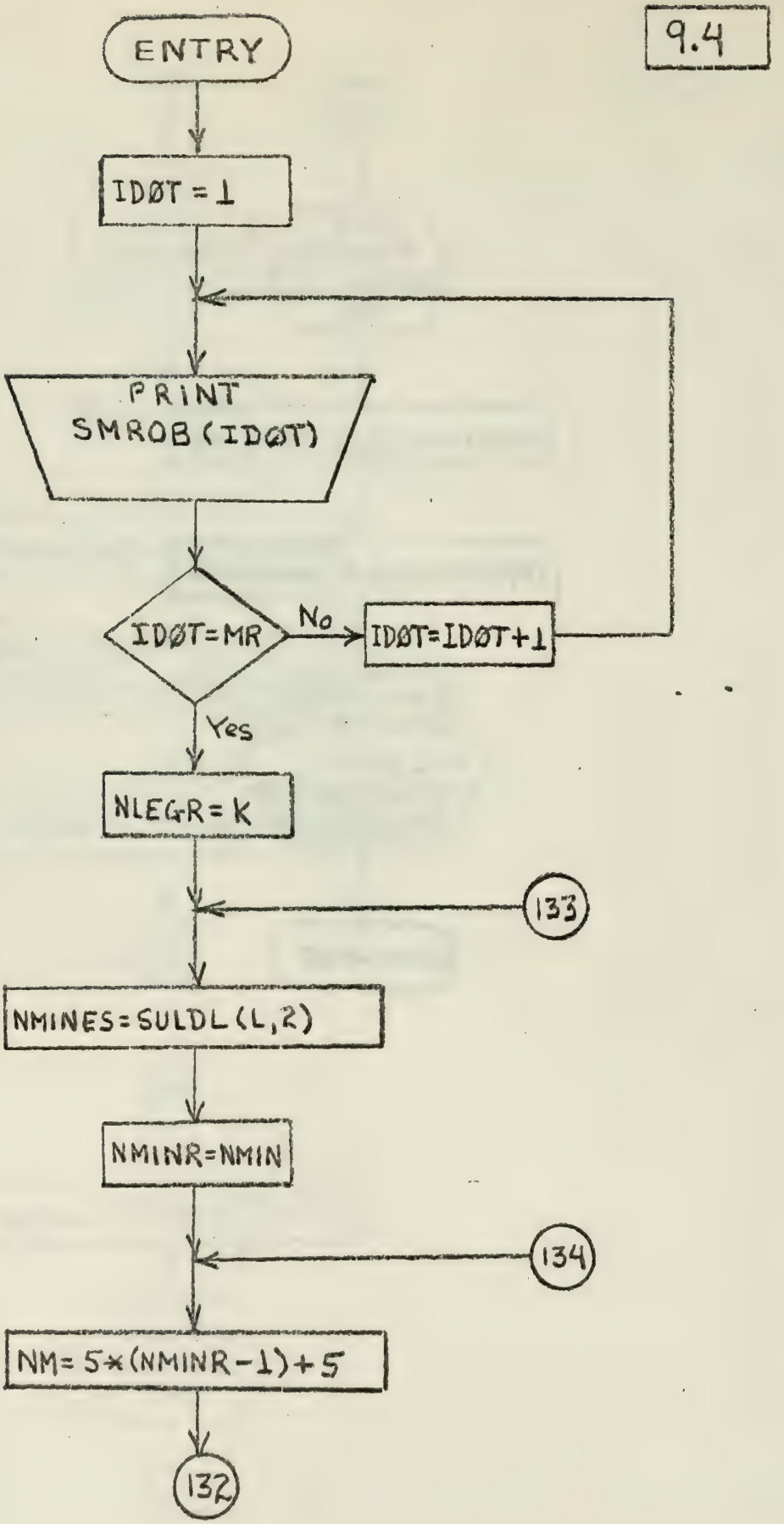

127
SUMCL(MT2,10) -RN<0
Yes.
128
No
OUTPUT(14)=1.
CALL RANU (IR,RN)
GYRUN=(HYP-RUN)*RN
CALL RANU (IR,RN)
GYPHI=6.28318*RN
XPOSIT=XPOSIT+RUN*DX/HYP+GYRUN*COSF(GYPHI)
YPOSIT=YPOSIT+RUN*DY/HYP+GYRUN*SINF(GYPHI)
131

9.3 CONTD

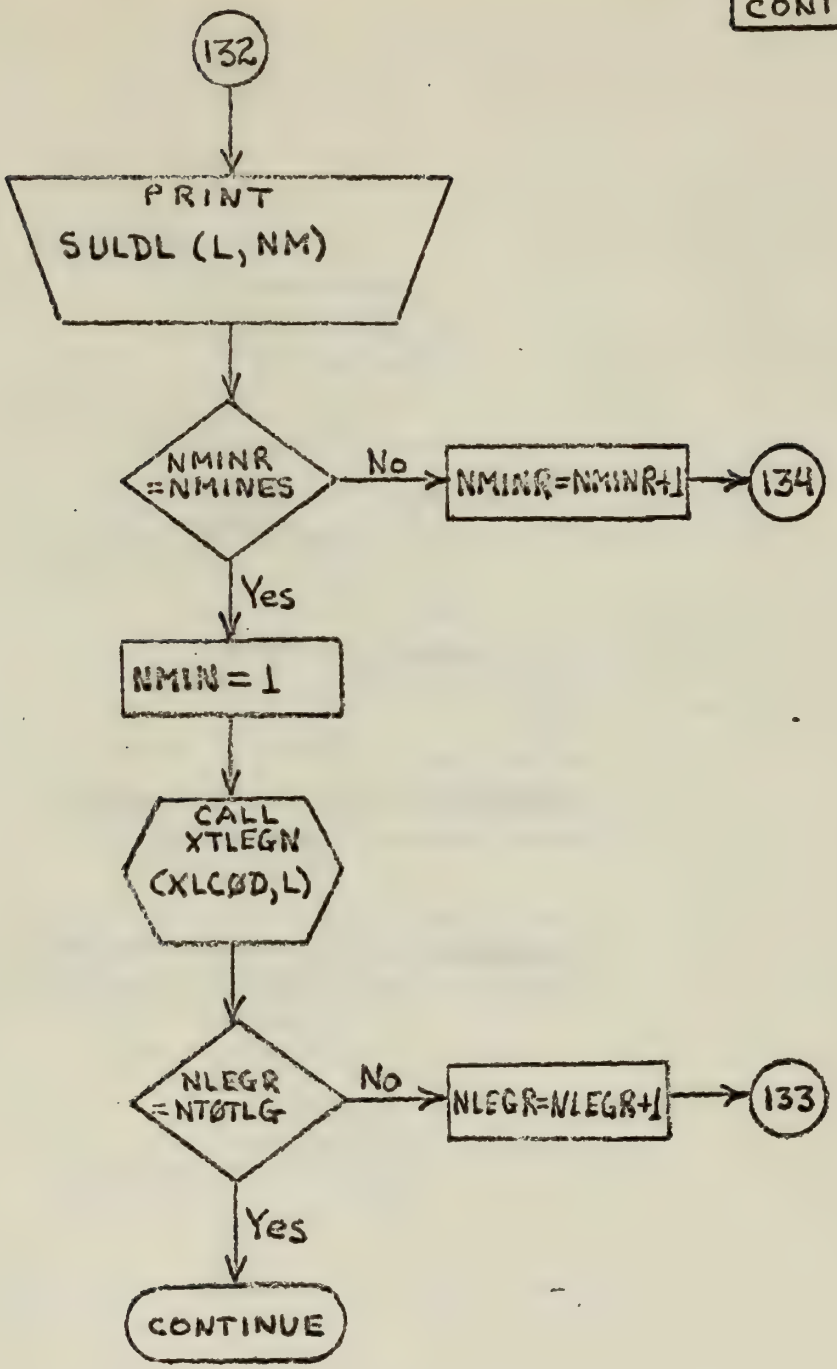

128
CALL RAN1 (IR,RN)
ØUTPUT(15)=1.
DRUN=(HYP-RUN)*RN+RUN
XPØSIT=XPØSIT + DRUN * DX/HYP
YPØSIT=YPØSIT + DRUN * DY/HYP
131
129
SIGMA = SUMSL(MT2,12) / 1.1744
CALL RNNØRM (SIGMA,XE, IR)
130

9.3 CONTD

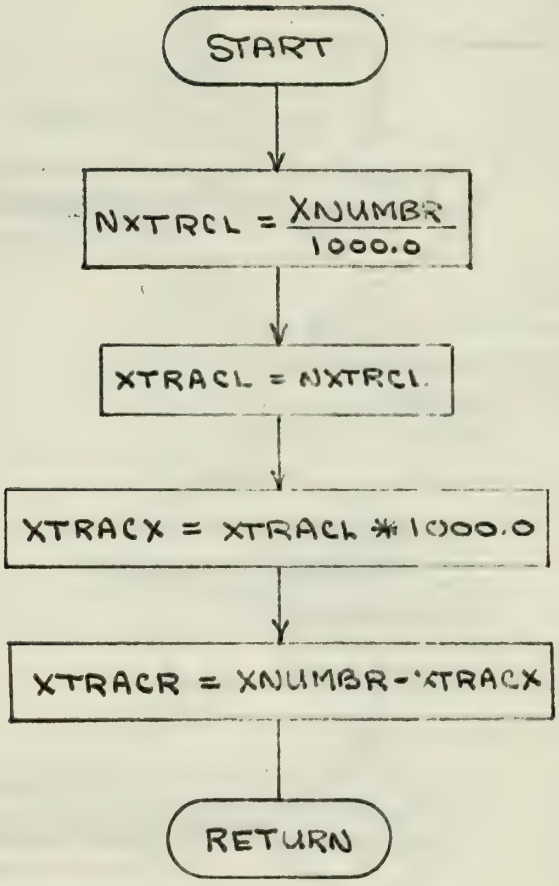

130
CALL RNNØRM (SIGMA, YE, RND)
XPØSIT=SULDL(L,NM+3)+XE
YPØSIT=SULDL(L,NM+4)+YE
131
CALL DEPTH (XPØSIT, YPØSIT, PHI, NMAX, MMAX, YMEANY, XMEANX, WD)
CONTINUE


ENTRY
IDØT = 1
PRINT
SMROB (IDØT)
IDØT=MR
No
IDØT=IDØT+1
Yes
NLEGR = K
133
NMINES = SULDL (L,2)
NMINR=NMIN
134
NM= 5*(NMINR-1)+5
132

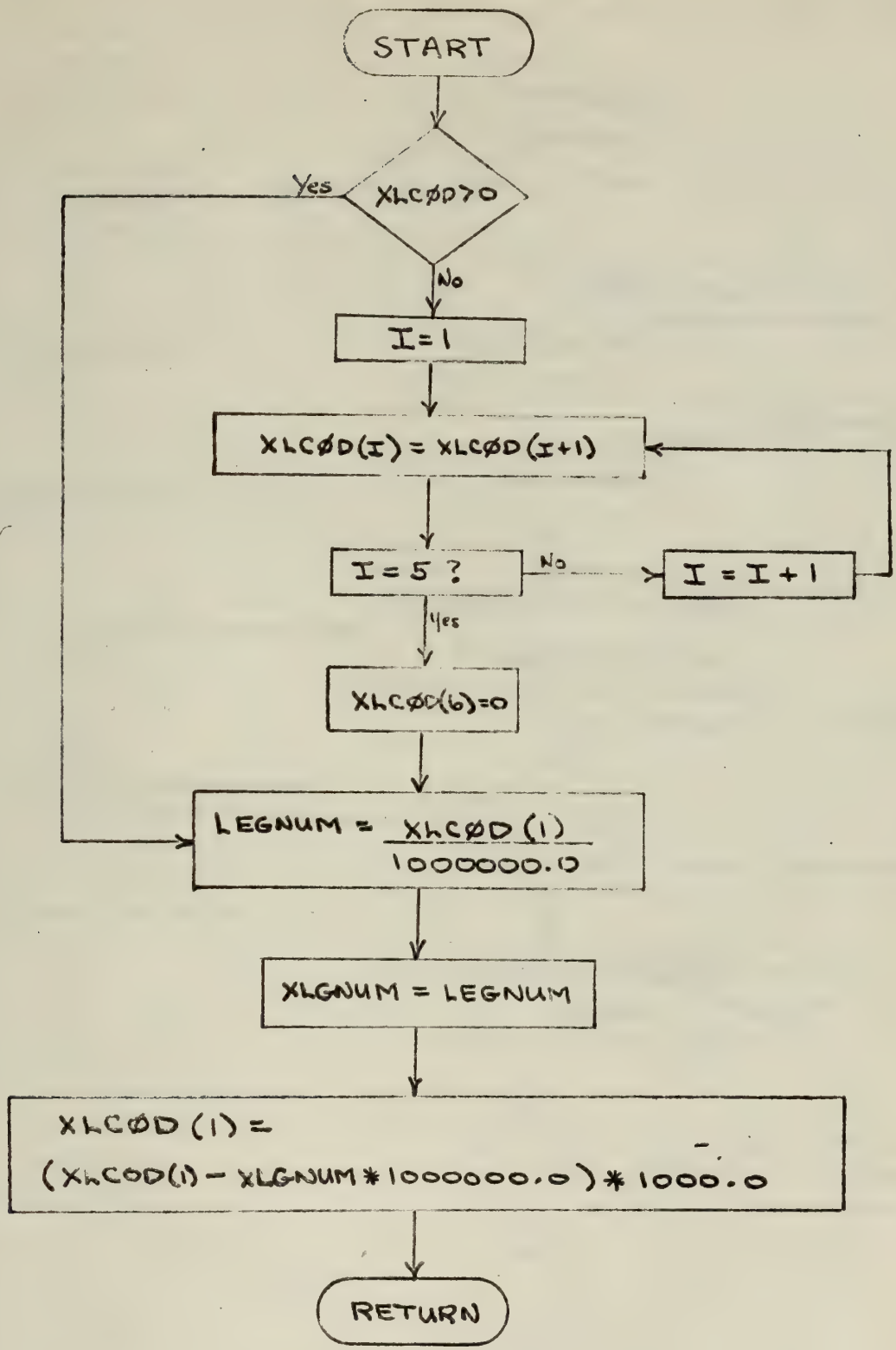
9.4
CONTD


132
PRINT
SULDL (L, NM)
NMINR
=NMINES
No
NMINR=NMINR+1
134
Yes
NMIN = 1
CALL
XTLEGN
(XLCØD, L)
NLEGR
=NTØTLG
No
NLEGR=NLEGR+1
133
Yes
CONTINUE

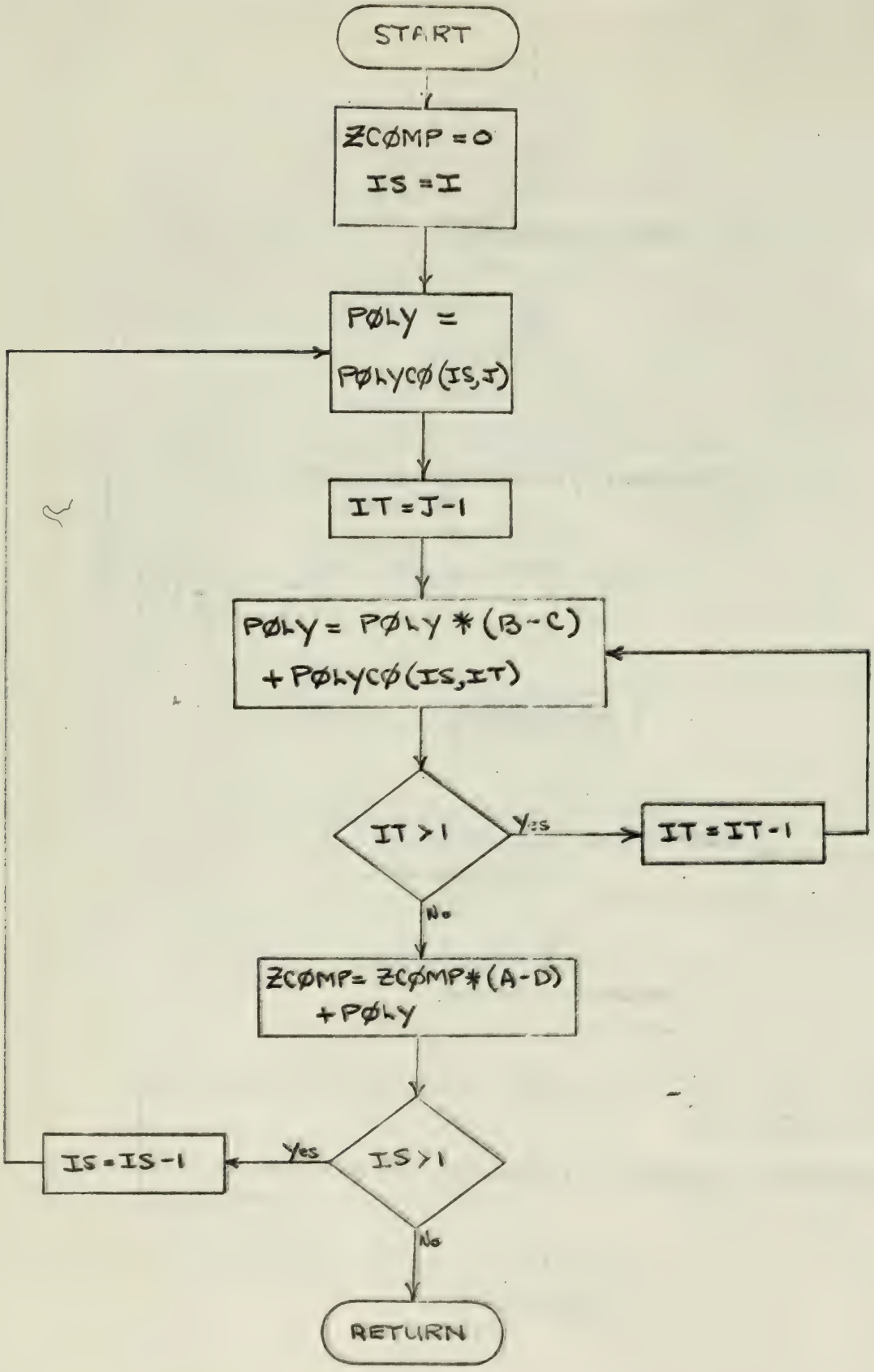
# SUBROUTINE XTRACT



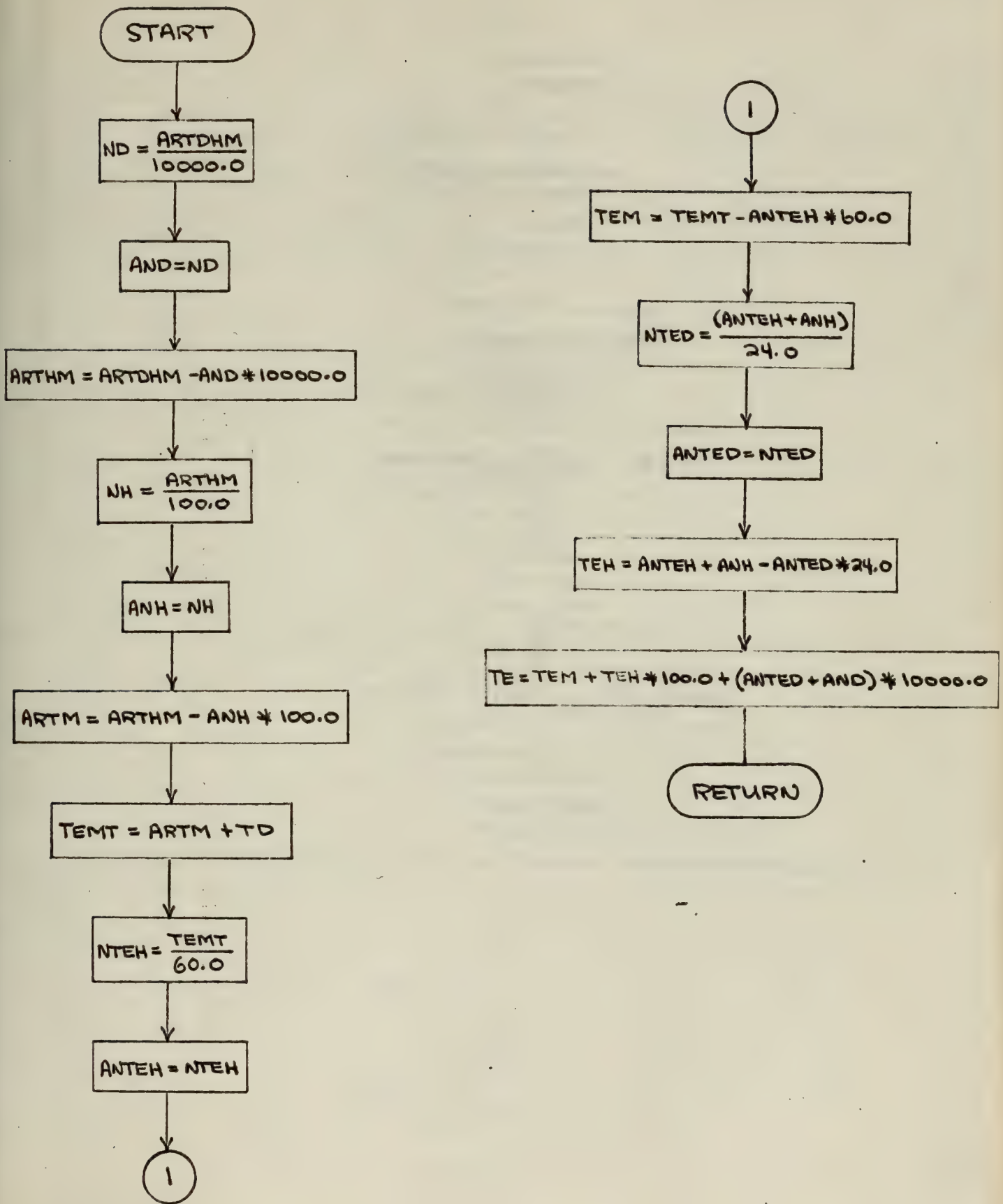
# SUBROUTINE XTLEGN



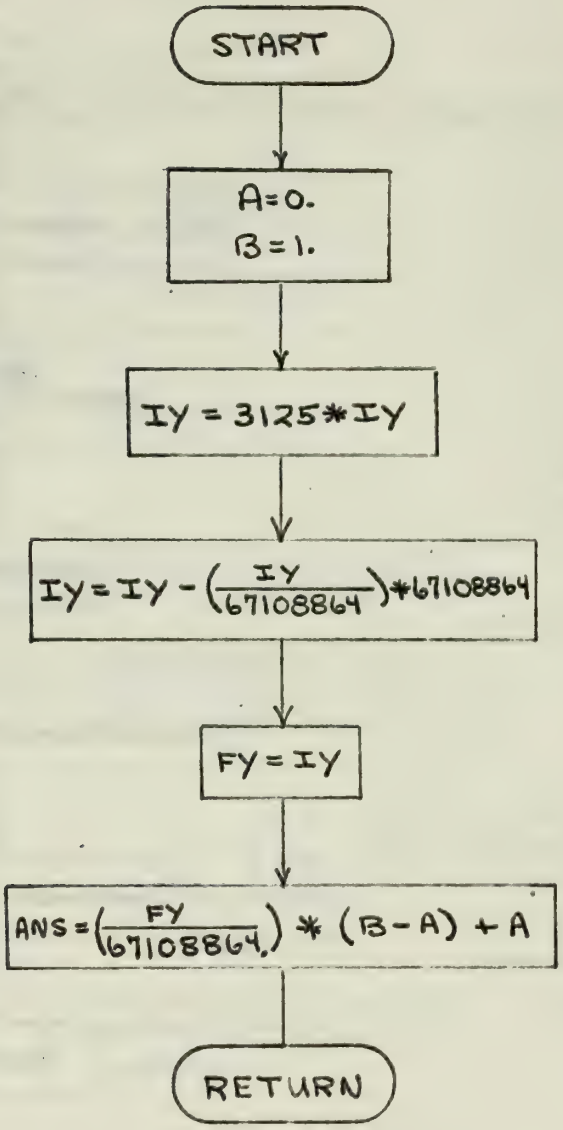
# SUBROUTINE DEPTH



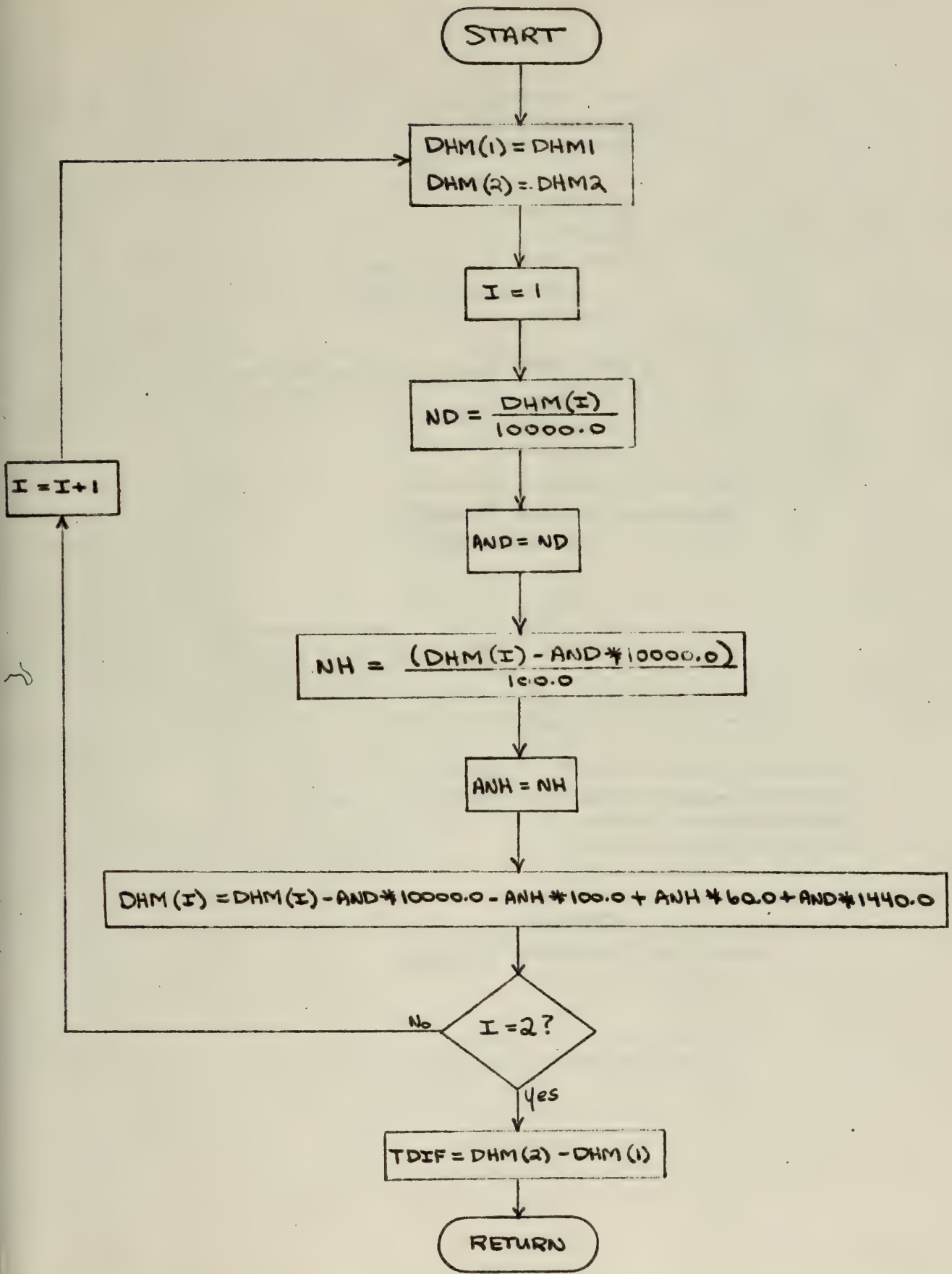
# SUBRØUTINE TIMCØN


START
ND = ARTDHM / 10000.0
AND=ND
ARTHM = ARTDHM -AND*10000.0
NH = ARTHM / 100.0
ANH = NH
ARTM = ARTHM - ANH * 100.0
TEMT = ARTM +TD
NTEH = TEMT / 60.0
ANTEH = NTEH
1
1
TEM = TEMT - ANTEH * 60.0
NTED = (ANTEH+ANH) / 24.0
ANTED = NTED
TEH = ANTEH + ANH - ANTED*24.0
TE = TEM + TEH * 100.0 + (ANTED + AND) * 10000.0
RETURN

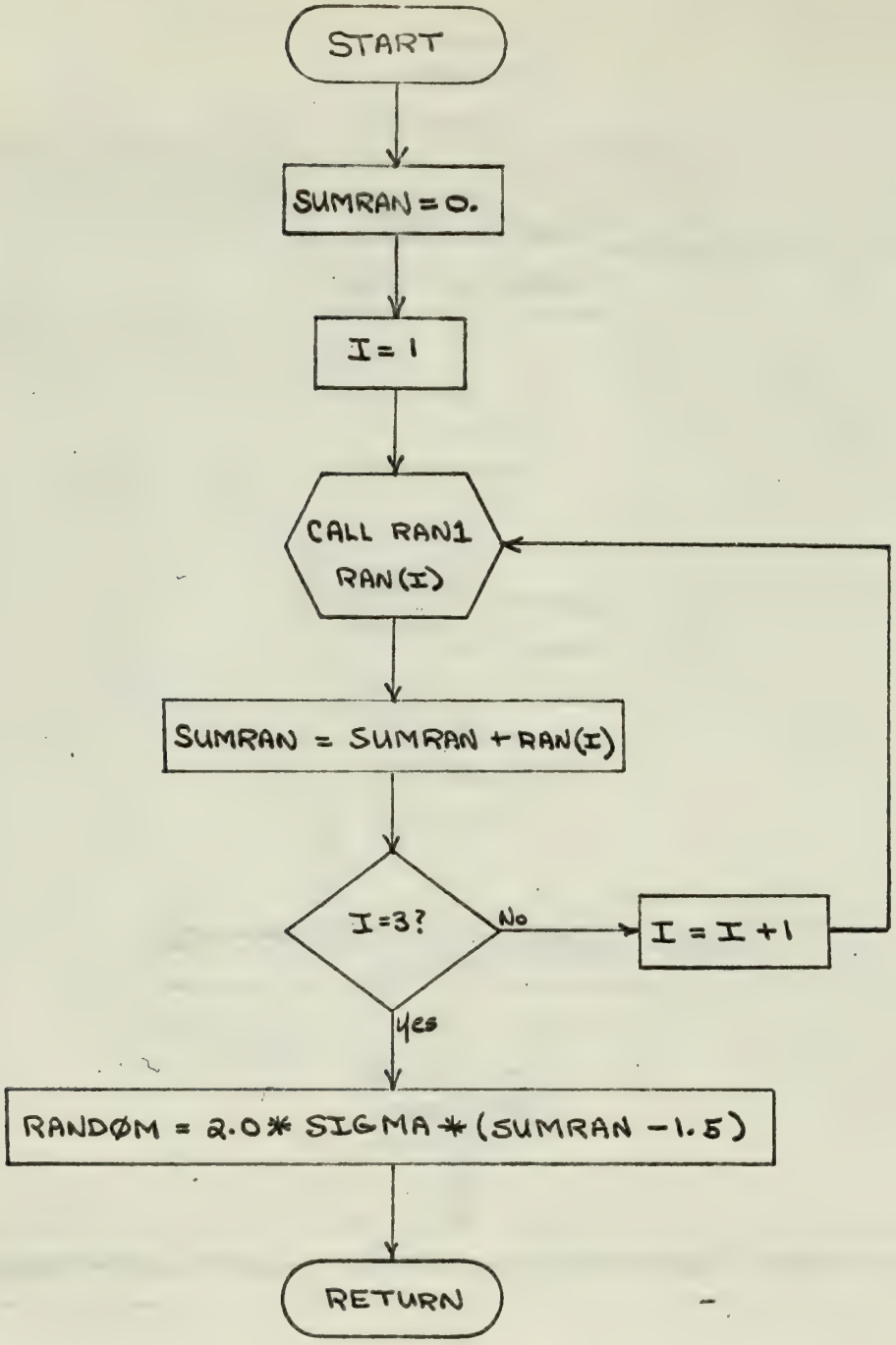
# SUBROUTINE RAN1



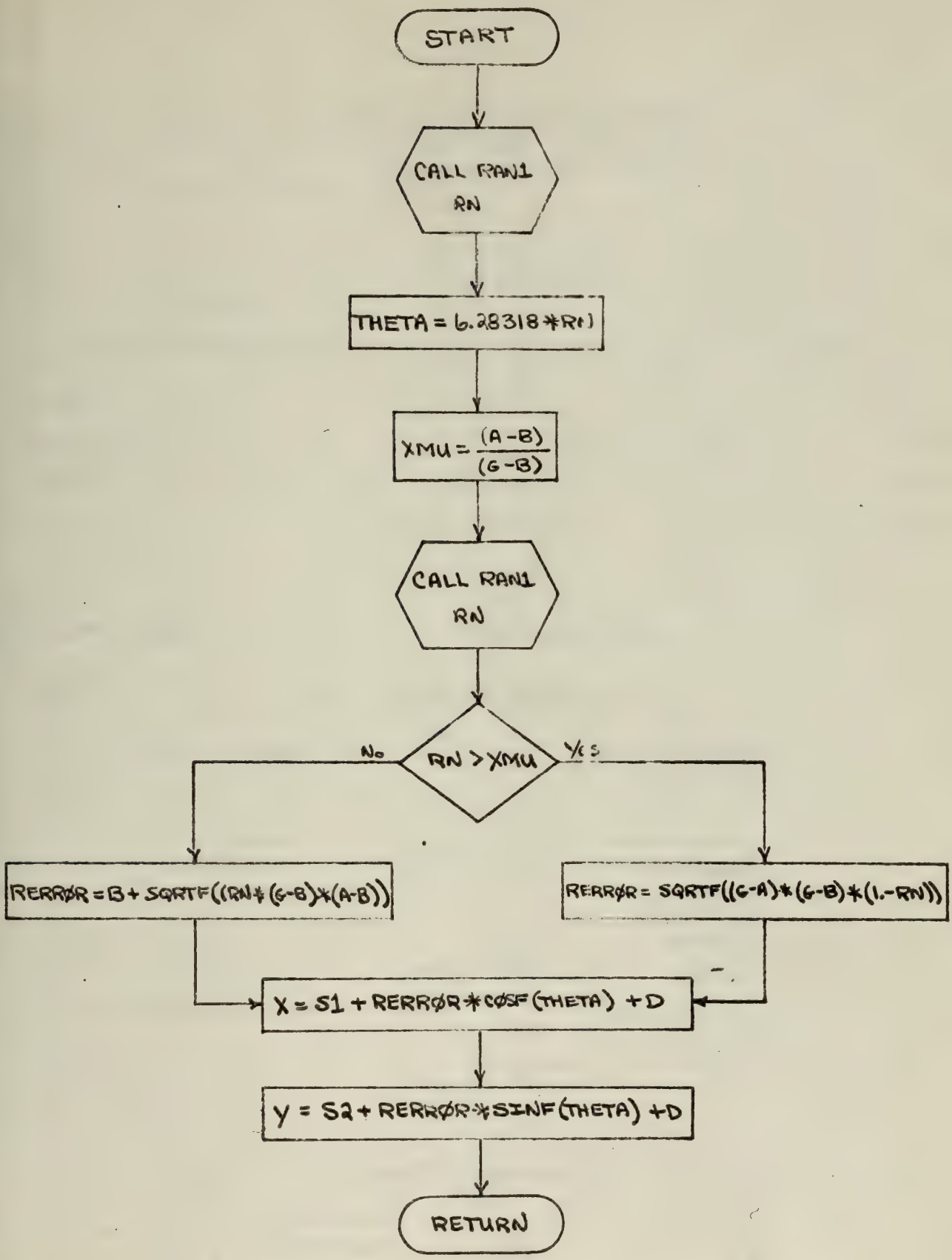
# SUBROUTINE TIMDIF



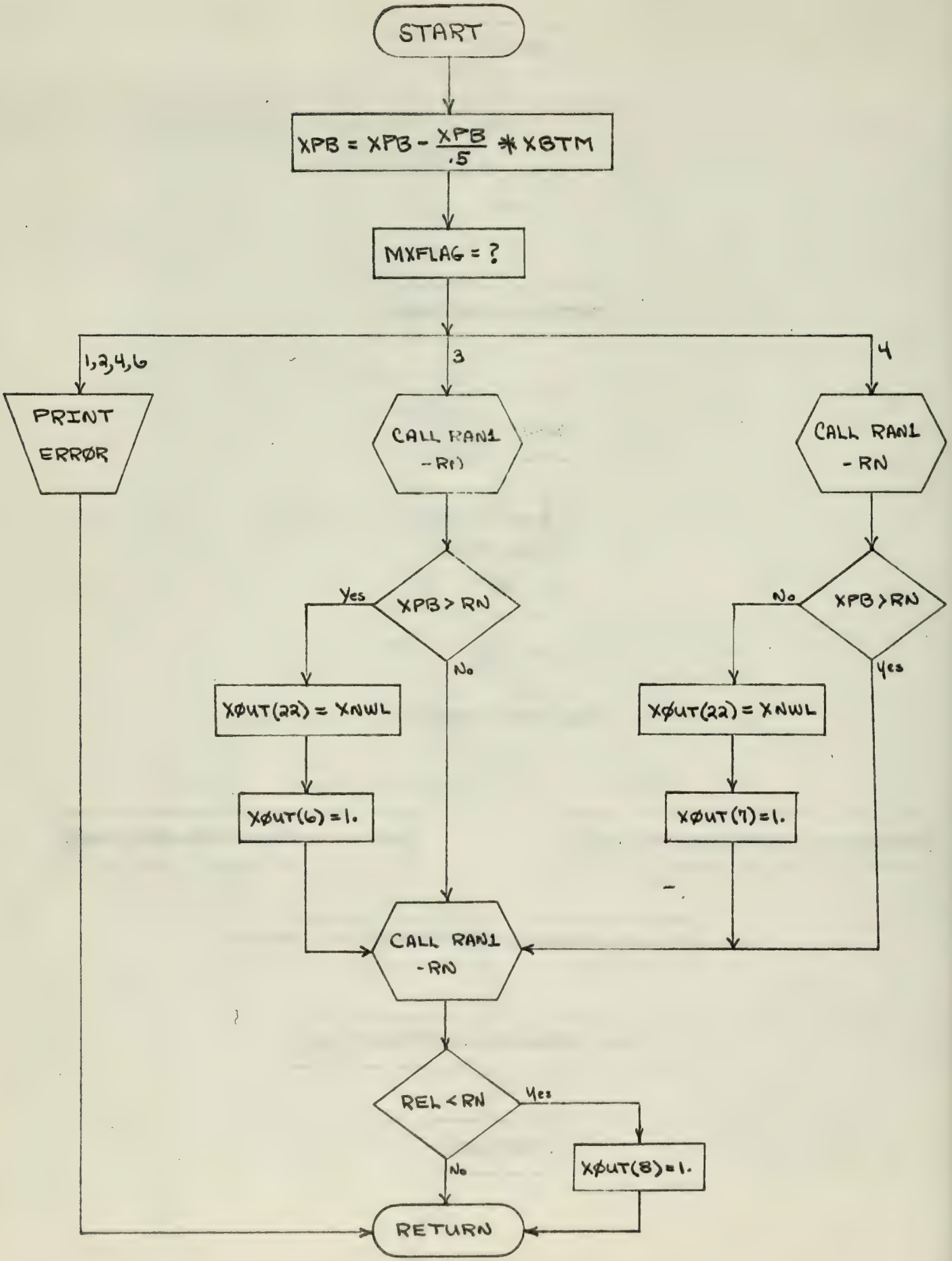
# SUBROUTINE RNORM



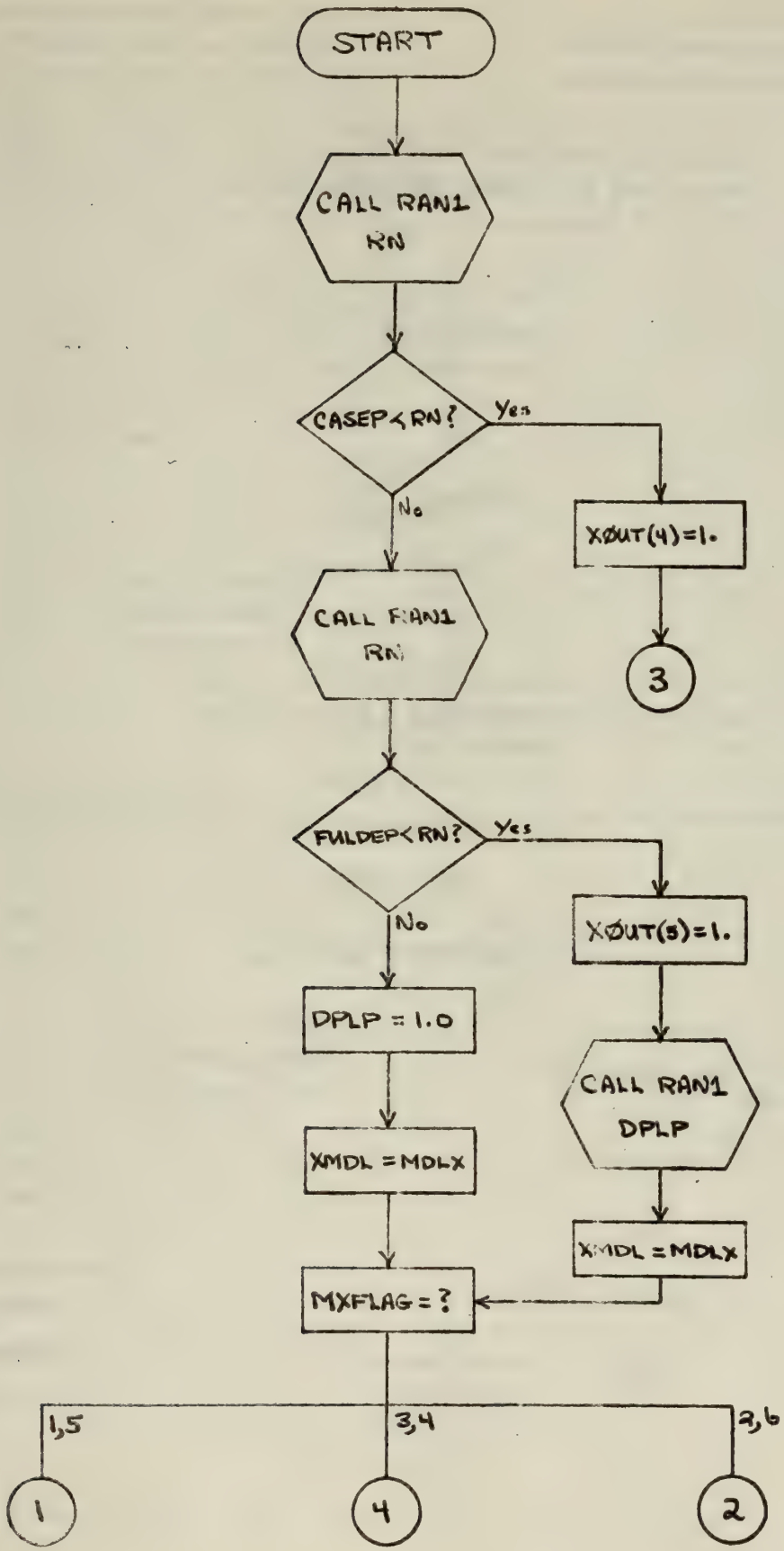
# SUBROUTINE POSIT

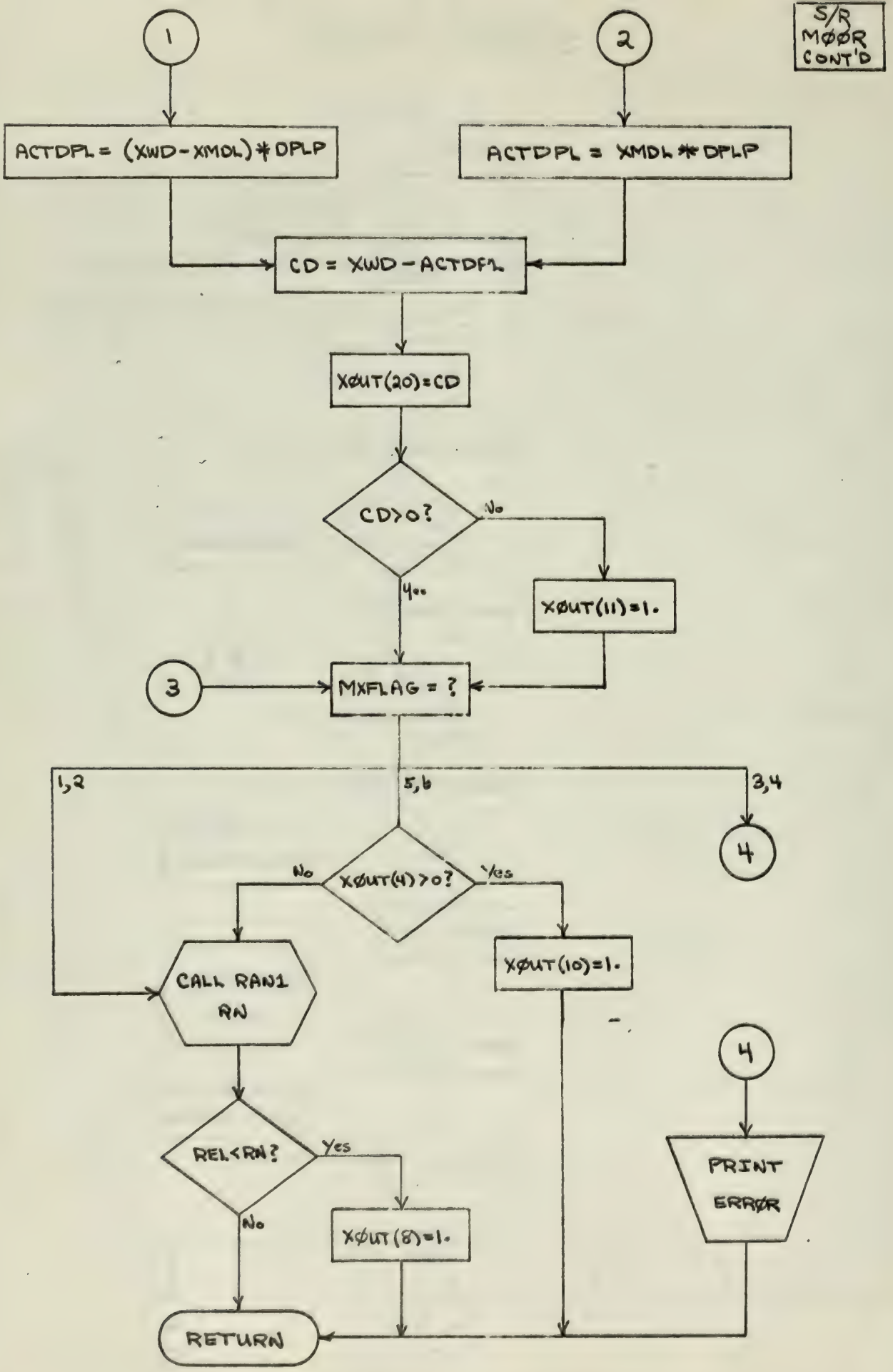


# SUBRØUTINE BØTTØM

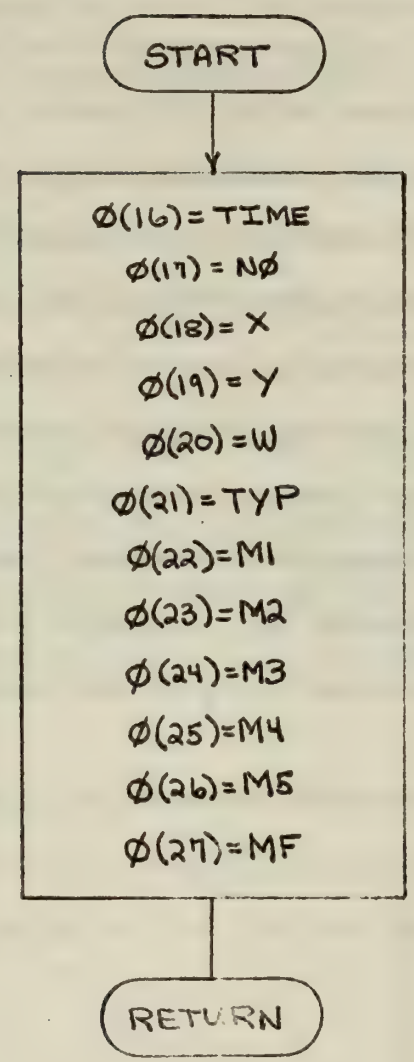


# SUBROUTINE MOSR




S/R MØØR CONT'D
1
2
ACTDPL = (XWD-XMDL)*DPLP
ACTDPL = XMDL*DPLP
CD = XWD - ACTDPL
XØUT(20)=CD
CD>0?
No
Yes
XØUT(11)=1.
3
MXFLAG = ?
1,2
5,6
3,4
4
No
XØUT(4)>0?
Yes
XØUT(10)=1.
CALL RAN1
RN
REL<RN?
Yes
No
XØUT(8)=1.
4
PRINT
ERRØR
RETURN

# SUBRØUTINE TAPO



APPENDIX VI

AIRCRAFT CIRCULAR PROBABLE ERROR

The navigational error technique for aircraft used in the mine delivery simulation involves the use of circular probable error (CEP), i.e., the 50 per cent probability circle. The decision to utilize CEP was based on a survey of publications concerning aerial minelaying and the observation that CEP was the most available form of data for aircraft navigational error. The data is generally tabulized with CEP as a function of aircraft altitude for selected ranges of airspeeds. Upon examination of these tables it is clear that CEP does not vary linearly with altitude, and it is here assumed that Altitude versus CEP has the form of the upper right half of a hyperbolic curve with its vertex at the origin as shown in the figure below:

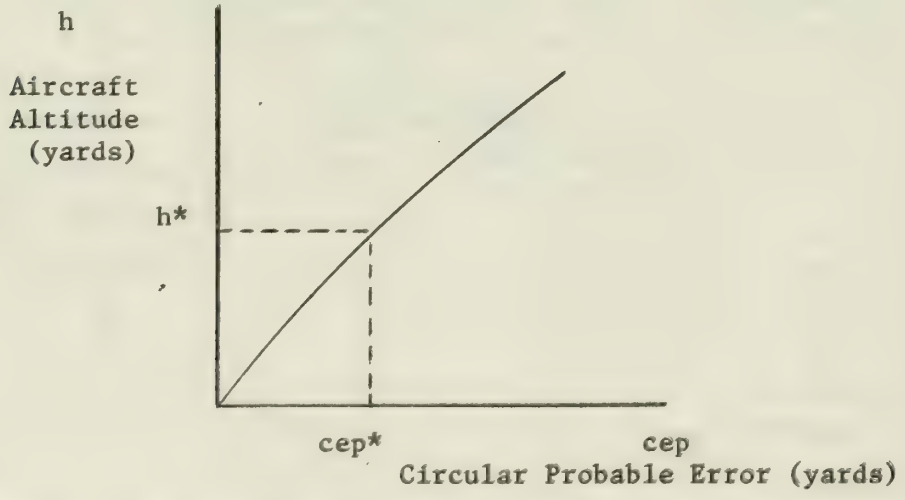



Figure 19. Aircraft Altitude versus Circular Probable Error.

The equation of the curve is:

$$h^2 = ( 1 - e^2 )( 1 - ( cep - 1 )^2 )$$

where h = aircraft altitude, e = geometric eccentricity, and cep = radius of Circular Probable Error.

Since the hyperbola passes through the origin, only one other point on the curve is required to determine the hyperbola's eccentricity. Using this fact and re-arranging the foregoing equation, the following formulation results:

$$e^* = \left( \left( 1 - ( cep^* - 1 )^2 - h^{*2} \right) / \left( 1 - ( cep^* - 1 )^2 \right) \right)^{\frac{1}{2}}$$

where $e^*$ = computed geometric eccentricity, $h^*$ = 1000 yards or 3000 feet of aircraft altitude, and $cep^*$ = CEP for 3000 feet of aircraft altitude from the Aircraft Mine Characteristics List in yards.

Again re-arranging the equation and using the calculated eccentricity, $e^*$, and the aircraft's minelaying altitude, H, from the Aircraft Characteristics List, the particular CEP is calculated:

$$CEP = \left( 1 - \left( H^2 / \left( 1 - e^{*2} \right) \right) \right)^{\frac{1}{2}} + 1$$

Utilizing this calculated CEP and the definition of CEP, the deviation of a normal distribution is determined:

$$Sigma = ( CEP / 1.1774 )$$

Assuming the normal distribution has mean zero, i.e., centered at the mine's intended position, and the above deviation, the X and Y

errors are generated by using the approximation[6]:

$$\text{Error} = \frac{\sum_{i=1}^{N} Z_i - N\,(E(Z))}{N^{\frac{1}{2}}\,(\text{dev.}(Z))}\ (\text{Sigma})$$

where Z is uniform (0,1).

The X and Y errors are algebraically added to the X and Y coordinates, respectively, of the mine's intended position to determine the final mine position.

[6]E. Parzen, Modern Probability and Its Application (New York: John Wiley and Sons, Inc., 1960), p.431.

## APPENDIX VII

## NAVIGATIONAL ERROR

A method of accurately simulating the problem of navigational error is required. Basically, a mine is to be layed at a precise geographical position, but the probability that the mine will be layed at exactly that position is zero due to navigational error. It is assumed that this error is Uniform on $(0,2\pi)$ in bearing, but the range distribution must be determined by some other means.

If sufficient data were available for every type of navigational equipment which might be encountered, giving the errors of these equipments under different operating conditions, it is conceivable that acceptable standard deviations for these equipments could be derived. Lacking this data, however,and desiring to provide navigational error information from as simple a routine as possible, the following technique has been adopted to approximate the range distribution. The technique is similar to that used to estimate completion times in a PERT system, but the development of the density function and the transformation of the Uniform (0,1) random variable is taken from the work of Underwood. [16]

The user is required to provide three inputs for each vehicle type navigational error. These three inputs are:

(1) BNE - an optimistic estimate of navigational error

(2) ANE - the most likely navigational error

(3) GNE - a pessimistic estimate of navigational error

As an example, take the case of a surface ship's navigational error. After considering all possible factors which could influ-

ence navigational accuracy for the ship type in the minelaying problem, the user may decide to input the following data -

BNE = 50 yds, ANE = 125 yds, GNE = 400 yds

The three estimates of navigational error define a triangular probability density function as shown below:

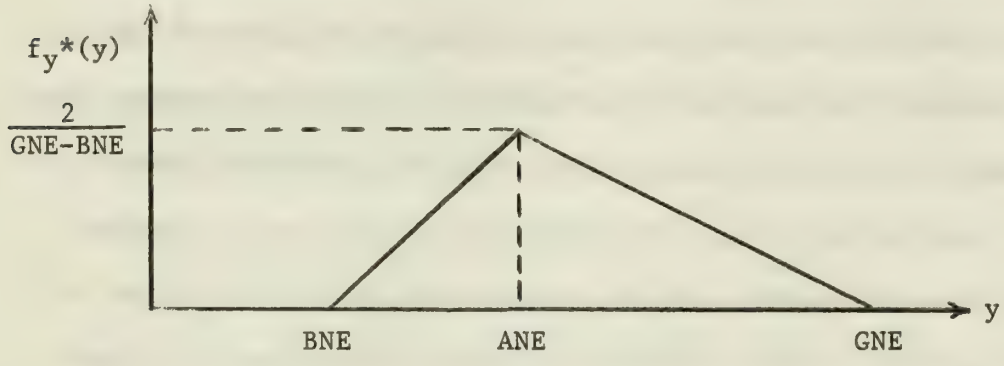



Figure 20. Triangular Density Function

Since the Distribution function of this density may be calculated by a simple formula, the transformation from a sample, u, drawn from a Uniform (0,1) distribution, to a sample from the desired triangular density, may be accomplished by a simple calculation.

The technique is illustrated here for the random variable, $y^*$, having the general triangular distribution with lower limit BNE, upper limit GNE, and mode ANE.

The density function of $y^*$ is:

$$f_y^*(y) = \frac{2(y-BNE)}{(GNE-BNE)(GNE-ANE)}, \qquad BNE \leq y \leq ANE$$

$$= \frac{2(GNE-y)}{(GNE-BNE)(GNE-ANE)}, \qquad ANE \leq y \leq GNE$$

$$= 0, \qquad \text{otherwise}$$

The distribution function of $y^*$ is:

$$F_{y^*}(y) = \int_{-\infty}^{y} f_{y^*}(z)\,dz$$

$$= \frac{(y-BNE)^2}{(GNE-BNE)(ANE-BNE)}, \qquad BNE \leq y \leq ANE$$

$$= \frac{ANE-BNE}{GNE-BNE} + \frac{ANE^2 - 2(GNE)(ANE)+2(GNE)y-y^2}{(GNE-BNE)(GNE-ANE)},$$

$$ANE \leq y \leq GNE$$

$$= 0, \quad y \leq BNE$$

$$= 1, \quad y \geq GNE$$

Now, let $u = F_{y^*}(y)$, and solve for the inverse function, $y = F_{y^*}^{-1}(u)$, and get

$$y = BNE + \sqrt{u(GNE-BNE)(ANE-BNE)}, \qquad 0 \leq u \leq \frac{ANE-BNE}{GNE-BNE}$$

$$= GNE - \sqrt{(GNE-ANE)(GNE-BNE)(1-u)}, \qquad \frac{ANE-BNE}{GNE-BNE} \leq u \leq 1$$

Therefore, if we draw a random number, u, from the Uniform (0,1) distribution, and determine y by means of the above function, then $y^*$ has the desired triangular distribution.

Using the above technique, the procedure for determining the final position of a mine is:

1. Locate the intended position (X,Y Co-ordinates).
2. Generate a random bearing.
3. Generate a random range.
4. Locate the final mine position on the random bearing at the random range from the intended position and convert to X, Y Co-ordinates.

Although there are undoubtedly more accurate methods of simulating navigational error, this method uses very little computer time and it is believed to be accurate enough for the purposes of this model. Additionally, the use of the three error estimates has the advantage of effectively eliminating any bias the user may introduce if only one estimate of navigational error were used.

## APPENDIX VIII

## SHIP ATTRITION

The determination of ship damage is based on the following assumptions:

1. The longer a ship is in the minelaying objective area the greater its probability of suffering some damage.

2. That this probability of a ship suffering some damage increases exponentially with time in the minelaying objective area.

3. That "damage" can be treated as the cumulative effect of all hits suffered by a ship.

4. The operational commander can give an estimate of the probability of some damage per ship type for the first half-hour period in the minelaying objective area, based on the expected defensive capabillty of the enemy.

The Probability of a Ship Suffering Some Damage as a function of time is defined as follows:

$$P(t) = e^{(1.4\alpha t)} - 1$$

This function is shown below for several values of alpha.

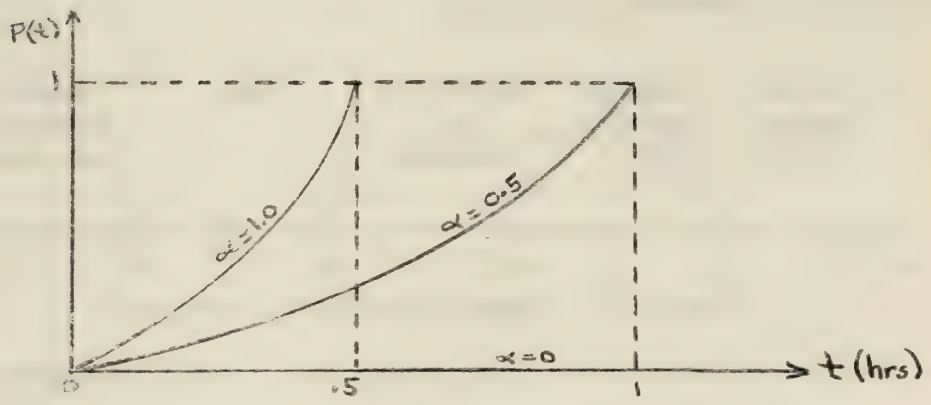



Figure 21. Ship Probability of Damage Function

The procedure for evaluating damage is as follows:

1. User supplies $P_u$ = P(some damage/ship type - first half-hour)

2. Compute

$$\alpha = \frac{\ln(P_u + 1)}{.7}$$

3. Compute time for P(t) = 1 and $\alpha$ as in (2) where

$$P(t) = e^{(1.4\alpha t)} - 1 \quad \text{for all } t,$$

$$T = \frac{1}{2}\left[\frac{\ln 2}{\ln(P_u + 1)}\right]$$

4. Compute total time ship under fire, $T_t$.

5. Compare T and $T_t$

   (a) If $T_t < T$, $P(t) = e^{(1.4\alpha t)} - 1$

   (b) If $T_t \geq T$, $P(t) = 1$

6. If $T_t < T$, compare a U(0,1) random number to $P(T_t)$ to determine if some damage is suffered. If $T_t \geq T$, some damage is suffered.

7. Compare a U(0,1) random number with $(0, T_t)$ to determine the time of damage if the ship suffers some damage.

If the ship is damaged, at the time of damage a uniform random number is compared to the following cumulative damage scale to determine the type of damage.

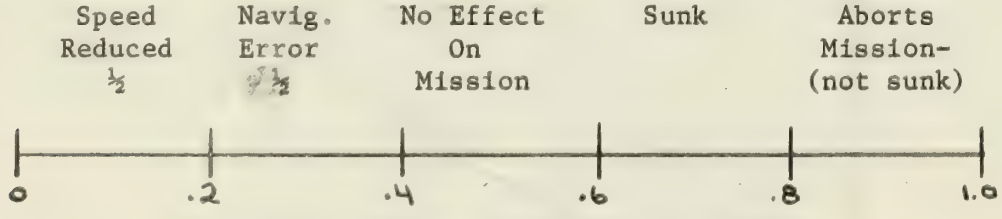



Figure 22. Damage Type Scale for Ships

Note: The increments on the above damage scale can be adjusted, as desired by the user, through a program change.

As an example, suppose the user provides $P_u = 0.2$ as the probability that a ship type will suffer some damage during its first half-hour in the minelaying objective area. Then,

$\alpha = 0.26$

$T = 1.91$ hours

Therefore, if the ship stays in the minelaying objective area for a period of time which is greater than or equal to 1.91 hours, its probability of damage is 1.0. Whereas, if the ship stays less than 1.91 hours its probability of suffering some damage is less than 1. If the ship stays for 1.5 hours,

$P(T_t) = 0.72$

## APPENDIX IX

## SUBMARINE DETECTION/ATTRITION ROUTINE

Because of the stealth and covertness unique to submarine mining operations, special handling of submarine detection and attrition was adopted. An examination of submarine and enemy capabilities led to the requirement that the user of the mine delivery simulation provide the following eight probabilities which represent the essence of the enemy action and submarine tactics possible:

1. Probability of detection - the chance that the enemy will detect the submarine while it is in the minefield objective area.
2. Probability of knowledge of detection given detection - the chance that the submarine will know he has been detected when actually detected.
3. Probability of thinking detected given no detection - the chance that the submarine considers itself detected when in fact detection has not occurred, e.g., a patrol craft enters the vicinity or an aircraft passes low overhead.
4. Probability of suspending operations given knowledge of detection - the chance or operational doctrine requiring that the submarine suspend operations if it is detected. Suspension of operations always takes place prior to attack or localization.
5. Probability of attempting return to the mining mission given operations have been suspended - the chance or

operational requirement that the submarine return to complete the mission, if it suspends operations.

6. Probability of attack given detection - the chance that the enemy will, and has the means to, press home an attack.
7. Probability of a successful attack given an attack - the chance that the enemy's attack will be a success, i.e., the submarine is sunk.
8. Probability of attempting return given an unsuccessful attack - the chance or operational requirement that the submarine return after surviving attack and evading.

By those probabilities where operational doctrine was indicated, the user is able by inputing zero or one to prescribe the mining mission doctrine for submarines.

Assuming that the marginals of the conditional probabilities and the probability of detection listed in the first paragraph are independent, and hence multiplicative in their intersections, the eight probabilities are combined to form 14 exhaustive and mutually exclusive events which are:

1. The submarine is not detected and does not think it is detected.
2. The submarine is not detected, thinks it is detected, and does not suspend operations.
3. The submarine is not detected, thinks it is detected, suspends operations, and does not attempt return.
4. Same as 3., but the submarine does attempt return.

5. The submarine is detected, knows it is detected, does not suspend operations, and is not attacked.
6. Same as 5., but the submarine is attacked, survives the attack, and does not return.
7. The submarine is detected, knows it is detected, does not suspend operations, is attacked, survives attack, and attempts return.
8. The submarine is detected, does not suspend operations, is attacked, and is sunk.
9. The submarine is detected, knows it is detected, suspends operations, and does not attempt return.
10. Same as 9., but the submarine attempts return.
11. The submarine is detected, does not know it is detected, and is not attacked.
12. The submarine is detected, does not know it is detected, is attacked, and is sunk.
13. The submarine is detected, does not know it is detected, is attacked, survives attack, and does not attempt return.
14. Same as 13., but the submarine does attempt return.

These 14 probabilities are considered distributed with segment lengths proportional to their computed values on the unit interval. A uniform (0,1) random number is compared to this interval to determine which detection and tactical action chain takes place.

In the instances where the submarine suspends operations and attempts return or is attacked and attempts return, the time out of the field is computed by one of two formulations based on the following assumptions:

1. The submarine will transit to beyond the 100 fathom curve in any evasive action.
2. If attacked, the submarine will spend more time in evasion than if he were only detected but not localized.

The time out of the field is computed in terms of the relative time of return (TRET) which for a submarine that has suspended operations is

$$TRET = DT + 2.\ (\ TD100\ ) + (\ TDIF - DT\ )(RN)$$

and for a submarine that has been attacked is

$$TRET = DT + 2.\ (\ TD100\ ) + 2.\ (\ TDIF - DT\ )(RN)$$

where DT is "dead time" in minutes, TD100 is the time it takes to transit to the 100 fathom curve, TDIF is the time of end of mission for submarines relative the submarine's time of arrival in minutes, and RN is a uniform (0,1) random number. The time out of the field for a submarine which has been attacked is greater than that for a submarine which has suspended operations by a factor of ( TDIF - DT )(RN).

The submarine returns at computed time and resumes its mining mission provided that the time of return does not exceed the End of Mission Time (TAM) set for submarines and that the mission was not already completed. The submarine upon its return is again subject to detection or attrition based on the same 14 probabilities and a new "dead time" computed from the time remaining to complete the mission. It is conceivable that the submarine may leave and return several times in one operation.

## APPENDIX X

## SELF-PROPELLED MINE FAILURES

In the Submarine Sequence of Events following the firing of a mine and the determination of the submarine's firing position, it must be determined if the mine is self-propelled. This is accomplished by testing the probability of failure for a self-propelled mine (PF) from the Submarine Mine Characteristics List for the mine type being layed. The value of PF should be zero for mine types that are not self-propelled. Having thus determined that the mine is self-propelled, a uniform random number is compared to PF to determine if a propulsion or control failure occurs. If a failure does not occur, then a circular probable error (CEP) about the mine's intended position is used to determine its final position.

If a failure does occur, the type of failure that takes place must be determined so that the failure characteristics may be considered in calculating the mine's final position. There are three possible types of failure, motor failure, steering failure, and depth control failure. These three failure types are represented by the three probabilities, probability of motor failure, probability of gyro failure, and the probability of depth control failure from the Submarine Mine Characteristics List, and are mutually exclusive and exhaustive. The three probabilities are considered distributed with segment lengths proportional to their values on the unit interval. A uniform random number is compared to this interval to determine which type of failure takes place.

For a motor failure the underlying concept is that the mine will come to rest at a random point along its intended path. This point is calculated by multiplying the directed distance from the submarine's firing position to the mine's intended position by a uniform random number and vectorially adding the resultant value to the submarine's firing position.

For a gyro or steering failure the underlying concept is that the mine will go erratic in steering control at a random point along its intended path, but its motor will continue to run. The mine then will come to rest at a random point in a circle centered at the point the mine went erratic and with a radius equal to length of the remaining intended path. The point at which the mine goes erratic is determined in the same manner as the final resting point for motor failure. To this point a distance calculated by multiplying a uniform random number by the length of the remaining intended path, and directed by $2\pi$ times a uniform random number, is vectorially added to determine the mine's final position.

For a depth control failure the concept is that the mine will go erratic in depth control at a random point along its intended path, and thereafter it will hunt about its set running depth in the vertical plane resulting in a reduced distance of advance along its intended path. Again, the point at which the mine goes erratic is determined in the same manner as the final resting point for motor failure. To this point is vectorially added the directed distance of the remaining intended path multiplied by a uniform

random number thus determining the mine's final position.

Having determined the mine's final position the sequence for determining the mine parameters is continued in a fashion identical to that for all other mines with one exception. In the case of motor failure the mine must run for a minimum distance, 500 yards as presently programmed, or it will not arm. This minimum arming distance may be changed by modification of the IF statement immediately prior to statement No. 575 of Program MINDEL.

## APPENDIX XI

## PROGRAM MINDEL1

Program MINDEL1 is an auxiliary program which has as its purpose the printing of the details regarding each mine layed and the compiling of selected statistics to summarize the results of several iterations. In addition to its primary purpose MINDEL1 prints the input to the main program, MINDEL.

At the outset of this program the increment size and the number of increments on both the X and Y axes are established by subroutine DELXY. The increment information is utilized in a later portion of the program to develop distributions of mine positions on the X and Y axes. Following the call of subroutine DELXY, all the input on magnetic tape from the main program is read and printed out in a tabular form with full captioning for easy reference. A sample is shown in Appendix XIII.

That which follows in program MINDEL1 hinges largely upon the structure of a 30-element array called OUTPUT. The elements of OUTPUT are:

1. Chute deployment indicator.
2. Water impact damage indicator.
3. Bottom impact damage indicator.
4. Case/anchor separation indicator.
5. Mooring cable deployment indicator.
6. Buried mine indicator.
7. Did not bury indicator.
8. Reliability indicator.

9. On the beach indicator.
10. Arming failure due to case/anchor separation failure indicator.
11. Mine watching indicator.
12. Arming failure due to short run indicator.
13. Motor failure indicator.
14. Gyro failure indicator.
15. Depth control failure indicator.
16. Time of lay.
17. Mine number.
18. X coordinate of mine's position.
19. Y coordinate of mine's position.
20. Case depth.
21. Mine type II.
22. NWL type.
23. Type of moor.
24. Ship counts.
25. Arming delay.
26. Sterile time.
27. MFLAG.
28. IR.
29. IREP.
30. Time of activation.

In the main program when an event was recorded, it was accomplished by setting the applicable indicator of the 15 indicator elements to zero or one. After determining the outcome of all the events in a particular mine's sequence, OUTPUT was written on magnetic tape.

The OUTPUT arrays are then read singly from tape by MINDEL1 and decoded until an end of file is encountered.

As each OUTPUT is read from magnetic tape, it is determined whether it is the beginning of an iteration or not. If so, all the elements are zero except the run number (IREP) and the random number generator initializing value (IR) which are then printed out. If not, those indicators of the first 15 elements which indicate the mine is inoperable, i.e., unreliable or damaged, are examined. If the mine is fully operable, the time of activation is computed by adding the arming delay to the time of lay and this time is stored in element number 30; otherwise, that element is zero. The full set of the first 15 elements is examined and indices to control the printing of adverse events, if any, are then set. Prior to any printout of the information regarding a mine, the X and Y coordinates are tested and counts are added to the corresponding increments of the X and Y axes. Also the applicable counts are added to the following statistics:

1. Number of mines layed.
2. Number of mines effective.
3. Number of mines buried.
4. Number of mines not buried.
5. Number of mines compromised.

All mines not damaged, unreliable, on the beach, or which failed to arm are considered effective. The number of mines buried (not buried) is the count of those mines for which burying (not burying) is undesirable. Compromised mines are those mines whose final

positions are on the beach or those moored mines which are watching. The mine number, time of lay, time of activation, X and Y coordinates, and case depth along with comment on adverse events are then printed for each mine layed. At the end of each iteration the total of the five statistics listed above is shown. Since the foregoing process generally exceeds the capacity of a standard output tape, subroutine STP was included to write an end file mark on the standard output tape and stop the program for changing the standard output tape at the computer operator's discretion.

Upon completion of processing all the OUTPUTs, a summary table of the five statistics along with the average of all the runs is presented. The count distribution on the X and Y axes is normalized, i.e., converted to a frequency distribution. The frequency distribution is then converted to a cumulative distribution. The values for both the frequency and cumulative distributions on the X and Y axes are then written on magnetic tape for use in program LSQRPLOT.

Sample output from program MINDEL1 appears in Appendix XIII. The program listing follows in this appendix. Preceding the program listing appear two pages describing the data deck for this program. The first of these two pages is of particular importance. It lists 15 cards containing alpha-numeric phrases which must appear in the order shown and as the first 15 cards of the data deck, since they are an integral part of the program's logic. On the succeeding page appears the description of the single additional data card required. As described in Appendices I, II, and

III, the card strip with letters is presented followed by columns of letter keys, FORTRAN specifications, variable names, and brief definitions.

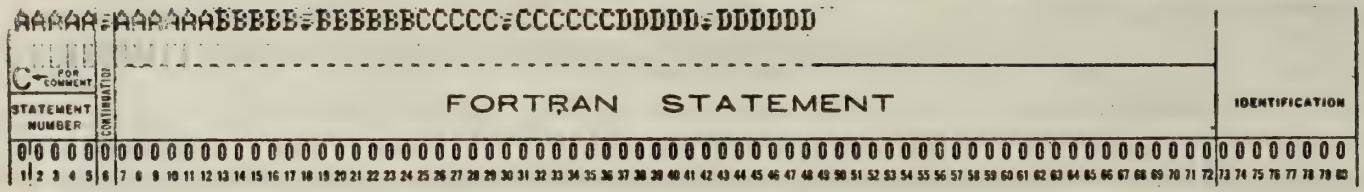
THE FOLLOWING PHRASES,PUNCHED UNDER FORMAT(7A8) MUST APPEAR AS THE FIRST FIFTEEN (15) DATA CARDS.

CHUTE DID NOT DEPLOY.
DAMAGED ON WATER IMPACT.
DAMAGED ON BOTTOM IMPACT.
CASE/ANCHOR SEPARATION FAILURE.
MOORING CABLE DID NOT DEPLOY FULLY.
BURIED.
DID NOT BURY.
UNRELIABLE.
ON THE BEACH.
DID NOT ARM DUE TO CASE/ANCHOR SEPARATION FAILURE.
MINE IS WATCHING.
DID NOT ARM DUE TO SHORT RUN.
MOTOR FAILURE.
GYRO FAILURE.
DEPTH FAILURE.

PROGRAM MINDEL1

INPUT DATA

1. DATA CARD




| | | | |
|---|---|---|---|
| A. | F12.6 | DELX | INCREMENT ON X-AXIS(YDS). |
| B. | F12.6 | DELY | INCREMENT ON Y-AXIS(YDS). |
| C. | F12.6 | XHALF | MEDIAN LENGTH OF X-AXIS(YDS). SAME AS XMEANX IN PROGRAM SURFIT. |
| D. | F12.6 | YHALF | MEDIAN LENGTH OF Y-AXIS(YDS). SAME AS YMEANY IN PROGRAM SURFIT. |

```
      PROGRAM MINDEL1                                                   00010
      DIMENSION ACLDL(8,31),SHLDL(7,70),SULDL(4,37),VAL(50,11),ACCL(10,6   00020
     1),SHCL(10,9),SUCL(10,17),ACMCL(24,14),SHMCL(10,8),SUMCL(8,13),BR(7   00030
     2,6),MDL(167,8),OUTPUT(30),PHRASE(105),KK(15),FXPTS(10),FYPTS(10)     00040
     3,X(500),Y(500),XSTAT(500),YSTAT(500),NSTAT(100,5),NTOT(5)            00050
      COMMON ACLDL,SHLDL,SULDL,VAL,ACCL,SHCL,SUCL,ACMCL,SHMCL,SUMCL,MDL,   00060
     1X,Y,XSTAT,YSTAT,NSTAT                                                00070
      READ 91,(PHRASE(J),J=1,105)                                           00090
   91 FORMAT(7A8)                                                           00100
      READ 511,DELX,DELY,XHALF,YHALF                                        00101
  511 FORMAT(4F12.6)                                                        00102
      CALL DELXY(X,Y,XSTAT,YSTAT,DELX,DELY,XHALF,YHALF,NUMX,NUMY,XINC,YI    00103
     1NC)                                                                   00104
      READ TAPE 2,XREF,YREF,FXPTS,FYPTS,ACLDL,SHLDL,SULDL,VAL,ACCL,SHCL,    00105
     1SUCL,ACMCL,SHMCL,SUMCL,BR,MDL,NV,NA,NAT,NACR,NACC,MACT,NSH,NSHT,NS    00106
     2HR,NSHC,MSHT,NSU,NSUT,NSUR,NSUC,MSUT,MINTOT,NF,TAM                    00107
      PRINT 5                                                               00110
    5 FORMAT(1H1,45X,29H INPUT TO MINE DELIVERY PHASE //)                   00120
      PRINT 600                                                             00121
  600 FORMAT(54X,11H FIELD DATA //)                                         00122
      PRINT 601,XREF,YREF                                                   00123
  601 FORMAT(28H THE X REFERENCE POSITION IS,F10.4,8H DEGREES // 28H THE    00124
     1 Y REFERENCE POSITION IS,F10.4,8H DEGREES //)                         00125
      PRINT 602,(FXPTS(I),FYPTS(I),I=1,NF)                                  00126
  602 FORMAT(48X,23H FIELD PERIMETER POINTS // 50X,19H XCOORD      YCOOR    00127
     1D   // (49X,F10.4,2X,F10.4/))                                        00128
      PRINT 6                                                               00130
    6 FORMAT(//50X,21H VEHICLE ARRIVAL LIST //)                             00140
      PRINT 7                                                               00150
    7 FORMAT(3X,4H TOA,4X,8H VEHICLE,2X,8H VEHICLE,3X,12H TOTAL NO.OF,2X    00160
     1,11H NO.OF BOMB,22X,14H LEGS ASSIGNED /13X,5H TYPE,3X,7H NUMBER,3X    00170
     2,14H LEGS ASSIGNED,3X,6H RACKS,28X,8H (CODED) //)                     00180
      PRINT 8,((VAL(I,J),J=1,11),I=1,NV)                                    00190
    8 FORMAT(F10.1,3X,F4.0,7X,F3.0,8X,F3.0,13X,F3.0,1X,F10.0,1X,F10.0,1X    00200
     1,F10.0,1X,F10.0,1X,F10.0,1X,F10.0)                                    00210
```

```
      PRINT 11                                                          00220
   11 FORMAT(////46X,30H AIRCRAFT CHARACTERISTICS LIST  //)             00230
      PRINT 12                                                          00240
   12 FORMAT(3X,13H VEHICLE TYPE,4X,11H XSIT SPEED,4X,10H M/L SPEED,4X,1 00250
     13H M/L ALTITUDE,4X,21H PROB OF MINE RELEASE,4X,22H PROB OF SURVIVA 00260
     2L OVER  /94X,15H OBJECTIVE AREA  //)                              00270
      PRINT 13,((ACCL(I,J),J=1,6),I=1,NAT)                              00280
   13 FORMAT(8X,F5.0,10X,F5.0,10X,F5.0,11X,F5.0,16X,F4.2,20X,F4.2)      00290
      PRINT 14                                                          00300
   14 FORMAT(1H1,45X,30H AIRCRAFT LEG DESCRIPTION LIST //)              00310
      PRINT 15                                                          00320
   15 FORMAT(41H LEG  NO.OF MINES  AIRCRAFT INITIAL POSIT,6X,9H MINE NO. 00330
     1,8X,23H MINE INTENDED POSITION,3X,24H AIRCRAFT TERMINAL POINT/4H N 00340
     2O.,2X,10H THIS  LEG,4X,18H XCOORD     YCOORD,28X,19H XCOORD      Y 00350
     3COORD,7X,20H XCOORD       YCOORD  //)                             00360
      DO100 I=1,NACR                                                    00370
      PRINT 16,(ACLDL(I,J),J=1,4)                                       00380
   16 FORMAT(F5.0,5X,F4.0,5X,F10.3,2X,F10.3)                            00390
      NSTOP = 3*XINTF(ACLDL(I,2)) + 4                                   00400
      PRINT 17,(ACLDL(I,K),K=5,NSTOP)                                   00410
   17 FORMAT(42X,F10.0,13X,F10.3,2X,F10.3/)                             00420
      NSTOP1 = NSTOP + 2                                                00430
      NSTOP2 = NSTOP + 3                                                00440
      PRINT 18,(ACLDL(I,J),J=NSTOP1,NSTOP2)                             00450
   18 FORMAT(91X,F10.3,3X,F10.3//)                                      00460
  100 CONTINUE                                                          00470
      PRINT 19                                                          00480
   19 FORMAT(1H1,46X,16H BOMB RACK ARRAY //)                            00490
      PRINT 20                                                          00500
   20 FORMAT(22H A/C BOMB RACK NO. ONE,13X,4H TWO,14X,6H THREE,14X,5H FO 00510
     1UR,14X,5H FIVE,14X,4H SIX/4H NO. //)                              00520
      DO101 I=1,NA                                                      00530
      PRINT 21,I,(BR(I,J),J=1,6)                                        00540
   21 FORMAT(I4,F19.0,F19.0,F19.0,F19.0,F19.0,F19.0)                    00550
  101 CONTINUE                                                          00560
```

```
      PRINT 22                                                          00570
   22 FORMAT(1H1,42X,35H AIRCRAFT MINE CHARACTERISTICS LIST //)         00580
      PRINT 23                                                          00590
   23 FORMAT(11H MINE TYPE ,3X,3H A ,3X,5H  B  ,3X,5H  C  ,3X,3H D ,3X,3 00600
     1H E ,3X,3H F ,3X,3H G ,3X,3H H ,3X,2H I,3X,2H J,3X,3H K ,3X,3H L   00610
     2,3X,3H M ,/3X,5H (II))                                            00620
      PRINT 24,((ACMCL(I,J),J=1,14),I=1,MACT)                           00630
   24 FORMAT(F6.0,7X,F4.2,3X,F5.0,3X,F5.0,3X,F3.2,3X,F3.2,3X,F3.2,3X,F3. 00640
     12,3X,F3.2,3X,F2.0,3X,F2.0,3X,F3.2,3X,F3.2,3X,F3.2)                00650
      PRINT 25                                                          00660
   25 FORMAT(29H A = PROB OF CHUTE DEPLOYMENT / 32H B = CEP FOR 3K FT,CH 00670
     1UTE DEPLOYS / 40H C = CEP FOR 3K FT,CHUTE DOES NOT DEPLOY / 46H D  00680
     2= PROB OF WATER IMPACT DAMAGE,CHUTE DEPLOYS / 54H E = PROB OF WATE 00690
     3R IMPACT DAMAGE,CHUTE DOES NOT DEPLOY / 93H F = PROB OF IMPACT DAM 00700
     4AGE ON THE HARDEST BOTTOM,GIVEN DEPTH LESS THAN 20FT AND CHUTE DEP 00710
     5LOYS / 97H G = PROB OF IMPACT DAMAGE ON HARDEST BOTTOM,GIVEN DEPTH 00720
     6 LESS THAN 20FT AND CHUTE DOES NOT DEPLOY / 38H H = PROB OF BURYIN 00730
     7G IN SOFTEST BOTTOM / 37H I = INDICATOR OF MOOR OR BOTTOM TYPE / 4 00740
     89H J = NWL TYPE IF BOTTOM MINE DOES NOT BURY/BURIES / 35H K = PROB 00750
     9 OF CASE/ANCHOR SEPARATION)                                       00760
      PRINT 26                                                          00770
   26 FORMAT(41H L = PROB OF FULL ANCHOR CABLE DEPLOYMENT / 68H M = PROB 00780
     1 MINE OPERATES,GIVEN IT HAS NOT BEEN DAMAGED (RELIABILITY)  //)   00790
      PRINT 250                                                         00800
  250 FORMAT(1H1,48X,25H SHIP CHARACTERISTIC LIST //)                   00810
      PRINT 251                                                         00820
  251 FORMAT(3X,13H VEHICLE TYPE,4X,11H XSIT SPEED,4X,10H M/L SPEED,5X,1 00830
     19H NAVIGATIONAL ERROR,5X,9H PROB  OF,2X,23H MAX     PROB OF DAMAGE 00840
     2/76X,32H MINE     TIME    FOR FIRST HALF/48X,59H LEAST  AVERAGE  G 00850
     3REATEST  RELEASE   DELAY     HOUR IN MLOA//)                      00860
      PRINT 252,((SHCL(I,J),J=1,9),I=1,NSHT)                            00870
  252 FORMAT(8X,F5.0,10X,F5.0,10X,F5.0,6X,F5.0,3X,F5.0,4X,F5.0,7X,F3.2,5 00880
     1X,F3.1,10X,F3.2)                                                  00890
      PRINT 267                                                         00900
  267 FORMAT(1H1,46X,26H SHIP LEG DESCRIPTION LIST//)                   00910
```

```
      PRINT 268                                                          00920
  268 FORMAT(41H LEG  NO.OF MINES    SHIP INITIAL POSIT  ,6X,9H MINE NO.  00930
     1,8X,23H MINE INTENDED POSITION,3X,24H   SHIP TERMINAL POINT  /4H N  00940
     20.,2X,10H THIS  LEG,4X,18H XCOORD     YCOORD,28X,19H XCOORD      Y  00950
     3COORD,7X,20H XCOORD       YCOORD  //)                             00960
      DO300 I=1,NSHR                                                     00970
      PRINT 269,(SHLDL(I,J),J=1,4)                                       00980
  269 FORMAT(F5.0,5X,F4.0,5X,F10.3,2X,F10.3)                             00990
      MSTOP = 3*XINTF(SHLDL(I,2)) + 4                                    01000
      PRINT 270,(SHLDL(I,K),K=5,MSTOP)                                   01010
  270 FORMAT(42X,F10.0,13X,F10.3,2X,F10.3/)                              01020
      MSTOP1 = MSTOP + 2                                                 01030
      MSTOP2 = MSTOP + 3                                                 01040
      PRINT 271,(SHLDL(I,J),J=MSTOP1,MSTOP2)                             01050
  271 FORMAT(91X,F10.3,3X,F10.3//)                                       01060
  300 CONTINUE                                                           01070
      PRINT 253                                                          01080
  253 FORMAT(////45X,30H SHIP MINE CHARACTERISTIC LIST//)                 01090
      PRINT 254                                                          01100
  254 FORMAT(66H MINE TYPE       A       B       C       D       E       01110
     1 F       G/3X,5H (II) )                                            01120
      PRINT 255,((SHMCL(I,J),J=1,8),I=1,MSHT)                            01130
  255 FORMAT(4X,F3.0,9X,F3.2,5X,F3.2,5X,F3.0,5X,F3.0,5X,F3.2,5X,F3.2,5X,  01140
     1F3.2)                                                             01150
      PRINT 266                                                          01160
  266 FORMAT(/64H A = PROB OF DAMAGE ON HARDEST BOTTOM,GIVEN DEPTH LESS  01170
     1THAN 20FT /38H B = PROB OF BURYING IN SOFTEST BOTTOM /37H C = INDI  01180
     2CATOR OF MOOR OR BOTTOM TYPE /48H D = NWL TYPE IF BOTTOM MINE DOES  01190
     3NOT BURY/BURIES /35H E = PROB OF CASE/ANCHOR SEPARATION /41H F = P  01200
     4ROB OF FULL ANCHOR CABLE DEPLOYMENT /67H G = PROB MINE OPERATES,GI  01210
     5VEN IT HAS NOT BEEN DAMAGED (RELIABILITY) //)                      01220
      PRINT 425                                                          01230
  425 FORMAT(1H1,45X,30H SUBMARINE CHARACTERISTIC LIST //)               01240
      PRINT 426                                                          01250
  426 FORMAT(41H VEHICLE  XSIT   M/L    NAVIGATION  ERROR,2X,74H PMR  FT  01260
```

```
   1DM  PD  PKD/D  PKD/ND  PSO/KD  PR/SO  PR/UA  PA/D  PSA/A    D(100)    01270
   2/42H  TYPE   SPEED  SPEED  BEST  AVG  GREATEST )                       01280
    PRINT 427,((SUCL(I,J),J=1,17),I=1,NSUT)                               01290
427 FORMAT(2X,F2.0,5X,F5.0,2X,F5.0,1X,F5.0,1X,F5.0,2X,F5.0,4X,F3.2,2X,    01300
   1F4.1,2X,F3.2,2X,F3.2,4X,F3.2,5X,F4.1,3X,F4.1,3X,F4.1,3X,F3.2,4X,F3    01310
   2.2,3X,F10.0)                                                          01320
    PRINT 428                                                             01330
428 FORMAT(/43H PMR    = PROB MINE IS READY AT FIRING TIME /37H FTDM      01340
   1 = FIRING TIME DELAY (MAXIMUM) /80H PKD/D  = PROB SUBMARINE HAS KN    01350
   2OWLEDGE OF DETECTION,GIVEN DETECTION HAS OCCURRED /73H PKD/ND = PR    01360
   3OB SUBMARINE THINKS DETECTION HAS OCCURRED,GIVEN NO DETECTION /73H    01370
   4 PSO/KD = PROB SUBMARINE SUSPENDS OPS,GIVEN IT HAS KNOWLEDGE OF DE    01380
   5TECTION /59H PR/SO  = PROB SUBMARINE RETURNS,GIVEN IT HAD SUSPENDE    01390
   6D OPS /57H PR/UA  = PROB SUBMARINE RETURNS,GIVEN UNSUCCESFUL ATTAC    01400
   7K /40H PA/D   = PROB OF ATTACK,GIVEN DETECTION /48H PSA/A  = PROB     01410
   8OF SUCCESSFUL ATTACK,GIVEN ATTACK /63H D(100) = DISTANCE FROM MINE    01420
   9FIELD PERIMETER TO 100 FATHOM CURVE //)                               01430
    PRINT 434,TAM                                                         01440
434 FORMAT(//52H DATE-TIME FOR COMPLETION OF SUBMARINE OPERATIONS IS,F     01450
   110.1 /)                                                               01460
    PRINT 320                                                             01470
320 FORMAT(1H1,44X,31H SUBMARINE LEG DESCRIPTION LIST //)                 01480
    PRINT 321                                                             01490
321 FORMAT(38H LEG  NO.OF MINES    SUB INITIAL POSIT,3X,73H MINE   SUB     01500
   1 FIRING POSIT    MINE INTENDED POSITION     SUB TERMINAL POINT/38H    01510
   2 NO.   THIS  LEG     XCOORD     YCOORD,4X,72H NO.   XCOORD    YCOO    01520
   3RD      XCOORD      YCOORD       YCOORD      YCOORD //)                01530
    DO400 I=1,NSUR                                                        01540
    PRINT 269,(SULDL(I,J),J=1,4)                                          01550
    KSTOP = 5*XINTF(SULDL(I,2)) + 4                                       01560
    PRINT 410,(SULDL(I,K),K=5,KSTOP)                                      01570
410 FORMAT(42X,F4.0,1X,F10.3,1X,F10.3,2X,F10.3,2X,F10.3/)                 01580
    KSTOP1 = KSTOP + 2                                                    01590
    KSTOP2 = KSTOP + 3                                                    01600
    PRINT 420, (SULDL(I,J),J=KSTOP1,KSTOP2)                               01610
```

```
420 FORMAT(95X,F10.3,1X,F10.3)                                          01620
400 CONTINUE                                                            01630
    PRINT 430                                                           01640
430 FORMAT(1H1,42X,35H SUBMARINE MINE CHARACTERISTIC LIST //)           01650
    PRINT 431                                                           01660
431 FORMAT(10H MINE TYPE,5X,68H A     B     C     D     E     F     G   01670
   1    H     I     J     K     L /2X,5H (II) )                        01680
    PRINT 432,((SUMCL(I,J),J=1,13),I=1,MSUT)                            01690
432 FORMAT(2X,F5.0,8X,F3.2,3X,F3.2,3X,F3.0,2X,F5.0,2X,F3.2,3X,F3.2,3X,  01700
   1F3.2,3X,F3.2,3X,F3.2,3X,F3.2,2X,F5.0,2X,F3.2 )                      01710
    PRINT 433                                                           01720
433 FORMAT(/37H A = PROB OF DAMAGE ON HARDEST BOTTOM /38H B = PROB OF   01730
   1BURYING IN SOFTEST BOTTOM /37H C = INDICATOR OF MOOR OR BOTTOM TYP  01740
   2E /49H D = NWL TYPE IF BOTTOM MINE DOES NOT BURY/BURIES /35H E = P  01750
   3ROB OF CASE/ANCHOR SEPARATION /41H F = PROB OF FULL ANCHOR CABLE D  01760
   4EPLOYMENT /49H G = PROB OF NORMAL RUN FOR A SELF-PROPELLED MINE /4  01770
   50H H = PROB OF MOTOR FAILURE,GIVEN FAILURE /39H I = PROB OF GYRO F  01780
   6AILURE,GIVEN FAILURE /40H J = PROB OF DEPTH FAILURE,GIVEN FAILURE   01790
   7/46H K = CEP FOR NORMAL RUN OF SELF-PROPELLED MINE /67H L = PROB M  01800
   8INE OPERATES,GIVEN IT HAS NOT BEEN DAMAGED (RELIABILITY) //)        01810
    PRINT 66                                                            01820
 66 FORMAT(1H1,45X,22H MINE DESCRIPTION LIST //)                        01830
    PRINT 67                                                            01840
 67 FORMAT(2X,5H MINE,5X,5H MINE,6X,5H MINE,6X,5H MINE,6X,7H ARMING,4X  01850
   1,8H STERILE,4X,5H SHIP,4X,8H TYPE OF /9H CASE NO.,2X,8H TYPE II,3X  01860
   2,7H TYPE I,2X,11H CASE DEPTH,4X,6H DELAY,5X,5H TIME,5X,7H COUNTS,4  01870
   3X,5H MOOR//)                                                        01880
    PRINT 68,((MDL(I,J),J=1,8),I=1,MINTOT)                              01890
 68 FORMAT(2X,I5,6X,I5,5X,I5,7X,I5,5X,I8,5X,I5,6X,I5,5X,I5)             01900
    DO 512 I=1,100                                                      01902
    DO 513 J = 1,5                                                      01904
513 NSTAT(I,J) = 0                                                      01906
512 CONTINUE                                                            01908
200 CALL STP                                                            01909
    READ TAPE 2,OUTPUT                                                  01910
```

```
      IF(EOF,2)9999,202                                                 01912
  202 IF((OUTPUT(2).EQ.0.).AND.(OUTPUT(3).EQ.0.).AND.(OUTPUT(8).EQ.0.).A 01914
     1ND.(OUTPUT(9).EQ.0.).AND.(OUTPUT(10).EQ.0.).AND.(OUTPUT(12).EQ.0.) 01916
     2)1000,201                                                         01918
 1000 JOUT = XINTF(OUTPUT(17))                                          01920
      ARMDEL = FLOATF(MDL(JOUT,5))                                      01922
      CALL TIMCON(OUTPUT(16),ARMDEL,OUTPUT(30))                         01924
  201 JJ = 0                                                            01926
      DO 500 II=1,15                                                    01930
  500 KK(II) = 0                                                        01940
      DO 30 I=1,15                                                      01950
      IF(OUTPUT(I))220,30,220                                           01960
  220 JJ = JJ + 1                                                       01970
      KK(JJ) = (I-1)*7 + 1                                              01980
   30 CONTINUE                                                          01990
      IF(JJ)31,32,31                                                    02000
   32 IF(OUTPUT(28))34,33,34                                            02010
   34 IR = XINTF(OUTPUT(28))                                            02020
      IREP = XINTF(OUTPUT(29))                                          02030
      IF(IREP.EQ.1)9998,9997                                            02032
 9997 MMM = IREP - 1                                                    02035
      PRINT 9996,(NSTAT(MMM,J),J=1,5)                                   02038
 9996 FORMAT(////13H  MINES LAYED,I5,17H  MINES EFFECTIVE,I4,14H  MINES 02040
     1BURIED,I4,18H  MINES NOT BURIED,I4,19H  MINES COMPROMISED,I4 /)    02044
 9998 PRINT 35,IREP,IR                                                  02048
   35 FORMAT(1H1,8H RUN NO.,I3,23H   IR FOR THIS RUN IS  ,I8 //)        02050
      PRINT 106                                                         02060
  106 FORMAT(51X,19H DETAILED NARRATIVE ///79H MINE NO.  TIME      TIME 02070
     1   XCOORD    YCOORD    CASE                    REMARKS/54H        02075
     2 LAYED    ACTIVE                          DEPTH /29X,25H (YDS)     (  02080
     3YDS)     (FT) //)                                                 02085
      GO TO 200                                                         02090
   33 PRINT 105,OUTPUT(17),OUTPUT(16),OUTPUT(30),(OUTPUT(J),J=18,20)    02100
      CALL COUNT(XSTAT,YSTAT,DELX,DELY,OUTPUT(18),OUTPUT(19),XHALF,YHALF 02103
     1)                                                                 02105
```

```
      CALL COMPIL(OUTPUT,NSTAT,IREP)                                    02108
      GO TO 200                                                         02110
   31 DO 40 J=1,15                                                      02120
      IF(KK(J))42,41,42                                                 02130
   42 GO TO(114,115,116,117,118,119,120,121,122,123,124,140,141,142,143) 02140
     1,J                                                                02150
  114 K1 = KK(1)                                                        02160
      K11 = K1 + 6                                                      02170
      GO TO 40                                                          02180
  115 K2=KK(2)                                                          02190
      K22=K2+6                                                          02200
      GO TO 40                                                          02210
  116 K3=KK(3)                                                          02220
      K33=K3+6                                                          02230
      GO TO 40                                                          02240
  117 K4=KK(4)                                                          02250
      K44=K4+6                                                          02260
      GO TO 40                                                          02270
  118 K5=KK(5)                                                          02280
      K55=K5+6                                                          02290
      GO TO 40                                                          02300
  119 K6=KK(6)                                                          02310
      K66=K6+6                                                          02320
      GO TO 40                                                          02330
  120 K7=KK(7)                                                          02340
      K77=K7+6                                                          02350
      GO TO 40                                                          02360
  121 K8=KK(8)                                                          02370
      K88 = K8+6                                                        02380
      GO TO 40                                                          02390
  122 K9=KK(9)                                                          02400
      K99=K9+6                                                          02410
      GO TO 40                                                          02420
  123 K10=KK(10)                                                        02430
      K100=K10+6                                                        02440
```

```
      GO TO 40                                                          02450
  124 K111=KK(11)                                                       02460
      K1110=K111+6                                                      02470
      GO TO 40                                                          02480
  140 K112=KK(12)                                                       02490
      K1120=K112+6                                                      02500
      GO TO 40                                                          02510
  141 K113=KK(13)                                                       02520
      K1130=K113 + 6                                                    02530
      GO TO 40                                                          02540
  142 K114=KK(14)                                                       02550
      K1140=K114+6                                                      02560
      GO TO 40                                                          02570
  143 K115=KK(15)                                                       02580
      K1150=K115+6                                                      02590
      GO TO 40                                                          02600
   41 GO TO(125,126,127,128,129,130,131,132,133,134,135,144,145,146,147) 02610
     1,J                                                                02620
  125 K11=-10                                                           02630
      K1 = 0                                                            02635
      GO TO 40                                                          02640
  126 K22=-10                                                           02650
      K2 = 0                                                            02655
      GO TO 40                                                          02660
  127 K33=-10                                                           02670
      K3 = 0                                                            02675
      GO TO 40                                                          02680
  128 K44=-10                                                           02690
      K4 = 0                                                            02695
      GO TO 40                                                          02700
  129 K55=-10                                                           02710
      K5 = 0                                                            02715
      GO TO 40                                                          02720
  130 K66=-10                                                           02730
      K6 = 0                                                            02735
```

```
      GO TO 40                                                          02740
  131 K77=-10                                                           02750
      K7 = 0                                                            02755
      GO TO 40                                                          02760
  132 K88=-10                                                           02770
      K8 = 0                                                            02775
      GO TO 40                                                          02780
  133 K99=-10                                                           02790
      K9 = 0                                                            02795
      GO TO 40                                                          02800
  134 K100 = -10                                                        02810
      K10 = 0                                                           02815
      GO TO 40                                                          02820
  135 K1110=-10                                                         02830
      K111 = 0                                                          02835
      GO TO 40                                                          02840
  144 K1120=-10                                                         02850
      K112 = 0                                                          02855
      GO TO 40                                                          02860
  145 K1130=-10                                                         02870
      K113 = 0                                                          02875
      GO TO 40                                                          02880
  146 K1140=-10                                                         02890
      K114 = 0                                                          02895
      GO TO 40                                                          02900
  147 K1150=-10                                                         02910
      K115 = 0                                                          02915
   40 CONTINUE                                                          02920
      PRINT 105,OUTPUT(17),OUTPUT(16),OUTPUT(30),(OUTPUT(J),J=18,20),(PH  02930
     1RASE(I),I=K1,K11),(PHRASE(I),I=K2,K22),(PHRASE(I),I=K3,K33),(PHRAS  02940
     2E(I),I=K4,K44),(PHRASE(I),I=K5,K55),(PHRASE(I),I=K6,K66),(PHRASE(I  02950
     3),I=K7,K77),(PHRASE(I),I=K8,K88),(PHRASE(I),I=K9,K99),(PHRASE(I),I  02960
     4=K10,K100),(PHRASE(I),I=K111,K1110),(PHRASE(I),I=K112,K1120),(PHRA  02970
     5SE(I),I=K113,K1130),(PHRASE(I),I=K114,K1140),(PHRASE(I),I=K115,K11  02980
     650)                                                               02985
```

```
  105 FORMAT(//2X,F4.0,2F10.1,2F10.3,1X,F7.2,4X,7A8,(/58X,7A8))                02990
      CALL COUNT(XSTAT,YSTAT,DELX,DELY,OUTPUT(18),OUTPUT(19),XHALF,YHALF       03000
     1)                                                                        03010
      CALL COMPIL(OUTPUT,NSTAT,IREP)                                           03020
      GO TO 200                                                                03030
 9999 PRINT 9996,(NSTAT(IREP,J),J=1,5)                                         03040
      PRINT 520                                                                03050
  520 FORMAT(1H1,50X,18HSUMMARY OF RESULTS //8X,61HRUN NO.      MINES LA       03060
     1YED     MINES EFFECTIVE     MINES BURIED ,43H     MINES NOT BURIED      03070
     2     MINES COMPROMISED //)                                              03080
      PRINT 505,(I,(NSTAT(I,J),J=1,5),I=1,IREP)                                03090
  505 FORMAT(8X,I5,10X,I4,14X,I4,15X,I4,17X,I4,16X,I4)                         03100
      CALL AGGREG(NSTAT,IREP,NTOT)                                             03110
      PRINT 510,(NTOT(J),J=1,5)                                                03120
  510 FORMAT(/8X,7HAVERAGE,9X,I4,14X,I4,15X,I4,17X,I4,16X,I4 /)                03130
      WRITE TAPE 3,NUMX,NUMY,X,Y,FXPTS,FYPTS,NF,XHALF,YHALF,NTOT(1)            03140
 9995 CALL NRMLZ(XSTAT,YSTAT,NUMX,NUMY)                                        03150
      WRITE TAPE 3,XSTAT,YSTAT                                                 03160
      CALL CMLTV ( XSTAT,YSTAT,NUMX,NUMY)                                      03170
      WRITE TAPE 3,XSTAT,YSTAT                                                 03180
      END                                                                      03190
```

```
      SUBROUTINE TIMCON (ARTDHM,TD,TE)                                  0320
      ND=ARTDHM/ 10000.0                                                0321
      AND = ND                                                          0322
      ARTHM = ARTDHM - AND * 10000.0                                    0323
      NH = ARTHM/100.0                                                  0324
      ANH = NH                                                          0325
      ARTM = ARTHM - ANH * 100.0                                        0326
       TEMT = ARTM + TD                                                 0327
      NTEH = TEMT / 60.0                                                0328
      ANTEH = NTEH                                                      0329
      TEM = TEMT - ANTEH * 60.0                                         0330
      NTED =(ANTEH +ANH)/24.0                                           0331
      ANTED=NTED                                                        0332
      TEH=ANTEH + ANH - ANTED*24.0                                      0333
      TE = TEM + TEH * 100.0 + (ANTED + AND) * 10000.0                  0334
      RETURN                                                            0335
      END                                                               0336
```

```
      SUBROUTINE DELXY(X,Y,XSTAT,YSTAT,DELX,DELY,XHALF,YHALF,NUMX,NUMY,X    0337
     1INC,YINC)                                                            0338
      DIMENSION X(500),Y(500),XSTAT(500),YSTAT(500)                        0339
      XDIV = (2.*XHALF)/DELX  $  YDIV = (2.*YHALF)/DELY                    0340
      NUMX = XFIXF(XDIV)+ 1   $   NUMY = XFIXF(YDIV) + 1                   0341
      X(1) = 0.  $  Y(1) = 0.  $ XSTAT(1) = 0.  $  YSTAT(1) = 0.           0342
      XINC = (2.*XHALF)/FLOATF(NUMX)  $  YINC = (2.*YHALF)/FLOATF(NUMY)    0343
      DO 1 I = 2,NUMX                                                      0344
      X(I) = X(I-1) + XINC                                                 0345
    1 XSTAT(I) = 0.                                                        0346
      DO 2 I = 2,NUMY                                                      0347
      Y(I) = Y(I-1) + YINC                                                 0348
    2 YSTAT(I) = 0.                                                        0349
      RETURN                                                               0350
      END                                                                  0351
```

```
      SUBROUTINE COUNT( XSTAT,YSTAT,DELX,DELY,XPOSIT,YPOSIT,XHALF,YHALF     0352
     1)                                                                     0353
      DIMENSION XSTAT(500),YSTAT(500)                                       0354
      IF((XPOSIT.LT.0.).OR.(YPOSIT.LT.0.).OR.(XPOSIT.GT.(2.*XHALF)).OR.(    0355
     1YPOSIT.GT.(2.*YHALF)))5,6                                             0356
    6 TX = XPOSIT / DELX                                                    0357
      TY = YPOSIT / DELY                                                    0358
      JX = XFIXF(TX)  $  XJ = FLOATF(JX)                                    0359
      JY = XFIXF(TY)  $  YJ = FLOATF(JY)                                    0360
      IF(DIMF(TX,XJ).GT..5)1,2                                              0361
    1 JX = JX + 1                                                           0362
    2 IF(DIMF(TY,YJ).GT..5)3,4                                              0363
    3 JY = JY + 1                                                           0364
    4 XSTAT(JX) = XSTAT(JX) + 1.0                                           0365
      YSTAT(JY) = YSTAT(JY) + 1.0                                           0366
    5 RETURN                                                                0367
      END                                                                   0368
```

```
      SUBROUTINE  NRMLZ( XSTAT,YSTAT,NUMX,NUMY)                          0369
      DIMENSION XSTAT(500),YSTAT(500)                                    0370
      XTOT = 0.                                                          0371
      DO 1 I=1,NUMX                                                      0372
    1 XTOT = XTOT + XSTAT(I)                                             0373
C     PRINT 20,XTOT                                                      0374
C  20 FORMAT(1H0,E15.8/)                                                 0375
      DO 2 I=1,NUMX                                                      0376
    2 XSTAT(I) = XSTAT(I) / XTOT                                         0377
      DO 3 I = 1,NUMY                                                    0378
    3 YSTAT(I) = YSTAT(I) / XTOT                                         0379
C     PRINT 10,(X(I),XSTAT(I),I=1,NUMX)                                  0380
C     PRINT 10,(Y(I),YSTAT(I),I=1,NUMY)                                  0381
C  10 FORMAT(1H1,(E15.8,10X,E15.8))                                      0382
      RETURN                                                             0383
      END                                                                0384
```

```
      SUBROUTINE COMPIL(OUTPUT,NSTAT,IREP)                                  0385
      DIMENSION OUTPUT(30),NSTAT(100,5)                                     0386
      NSTAT(IREP,1) = NSTAT(IREP,1) + 1                                     0387
      SUMO = OUTPUT(2)+ OUTPUT(3) + OUTPUT(8) + OUTPUT(9) + OUTPUT(10)      0388
     1 + OUTPUT(11) +OUTPUT(12)                                             0389
      IF(SUMO)1,1,2                                                         0390
    1 NSTAT(IREP,2) = NSTAT(IREP,2) + 1                                     0391
    2 IF(OUTPUT(6))4,4,3                                                    0392
    3 NSTAT(IREP,3) = NSTAT(IREP,3) + 1                                     0393
    4 IF(OUTPUT(7))6,6,5                                                    0394
    5 NSTAT(IREP,4) = NSTAT(IREP,4) + 1                                     0395
    6 IF(OUTPUT(9))8,8,7                                                    0396
    8 IF(OUTPUT(11))9,9,7                                                   0397
    7 NSTAT(IREP,5) = NSTAT(IREP,5) + 1                                     0398
    9 RETURN                                                                0399
      END                                                                   0400
```

```
      SUBROUTINE AGGREG(NSTAT,IREP,NTOT)                              0401
      DIMENSION NSTAT(100,5),NTOT(5)                                  0402
      DO 1 I = 1,5                                                    0403
      NTOT(I) = 0      .                                              0404
      DO 2 J = 1,IREP                                                 0405
    2 NTOT(I) = NTOT(I) + NSTAT(J,I)                                  0406
      NTOT(I) = NTOT(I)/IREP                                          0407
    1 CONTINUE                                                        0408
      RETURN                                                          0409
      END                                                             0410
```

```
      SUBROUTINE CMLTV ( XSTAT,YSTAT,NUMX,NUMY)                          0411
      DIMENSION XSTAT(500),YSTAT(500)                                    0412
      DO 1 I=2,NUMX                                                      0413
    1 XSTAT(I) = XSTAT(I) + XSTAT(I-1)                                   0414
      XSTAT(1)=0.                                                        0415
      DO 2 I=2,NUMY                                                      0416
    2 YSTAT(I)= YSTAT(I) + YSTAT(I-1)                                    0417
      YSTAT(1)=0.                                                        0418
C     PRINT 10,(XSTAT(I),YSTAT(I),I=1,801)                               0419
C  10 FORMAT(1H1,(E15.8,10X,E15.8))                                      0420
      RETURN                                                             0421
      END                                                                0422
```

```
      SUBROUTINE EFM                                                    0423
      END FILE 51                                                       0424
      RETURN                                                            0425
      END                                                               0426
```



```
      SUBROUTINE STP

          IDENT     STP                                                 0427
          ENTRY     STP                                                 0428
          EXT       EFM                                                 0429
STP       SLJ       **                                                  0430
          ENI       0                                                   0431
+         SLJ    1  AA                                                  0432
          SLJ       BB                                                  0433
AA        RTJ       EFM                                                 0434
          ENI       0                                                   0435
+         SLS       STP                                                 0436
BB        SLJ       STP                                                 0437
          END                                                           0438
```

# APPENDIX XII

## PROGRAM LSQRPLOT

On the pages which follow in this appendix is the program listing for program LSQRPLOT and is included here to demonstrate how the distribution information provided by program MINDEL1 may be utilized. The subroutines DRAW and LSCFWOP were copied from the U. S. Naval Postgraduate School Computer Facility Library.

```
      PROGRAM LSQRPLOT                                                  00010
      DIMENSION X(500),Y(500),XSTAT(500),YSTAT(500),FXPTS(10),FYPTS(10),00020
     1ITITLE(12)                                                        00025
      COMMON/BLO/X,Y,XSTAT,YSTAT                                        00030
      REWIND 2                                                          00040
      READ TAPE 2,NUMX,NUMY,X,Y,FXPTS,FYPTS,NF,XMEANX,YMEANY,N          00050
      READ TAPE 2,XSTAT,YSTAT                                           00060
      KK = 3                                                            00070
      ITITLE(1) = 8H                                                    00080
      ITITLE(2) = 8H   KRUMM                                            00090
      ITITLE(3) = 8H,T.G. AN                                            00100
      ITITLE(4) = 8HD SNYDER                                            00110
      ITITLE(5) = 8H,S.V.                                               00120
      ITITLE(6) = 8H                                                    00130
      ITITLE(7) = 8H     MIN                                            00140
      ITITLE(8) = 8HE FREQUE                                            00150
      ITITLE(9) = 8HNCY DIST                                            00160
      ITITLE(10) = 8HRIBUTION                                           00170
      ITITLE(11) = 8H ON X-AX                                           00180
      ITITLE(12) = 8HIS                                                 00190
      CALL DRAW(NUMX,X,XSTAT,0,0,4H    ,ITITLE,0.,0.,0,0,0,0,8,8,1,LAST)00200
      ITITLE(11) = 8H ON Y-AX                                           00210
      CALL DRAW(NUMY,Y,YSTAT,0,0,4H    ,ITITLE,0.,0.,0,0,0,0,8,8,1,LAST)00220
      READ TAPE 2,XSTAT,YSTAT                                           00230
      ITITLE(7) = 8H     MIN                                            00240
      ITITLE(8) = 8HE CUMULA                                            00250
      ITITLE(9) = 8HTIVE DIS                                            00260
      ITITLE(10) = 8HTRIBUTIO                                           00270
      ITITLE(11) = 8HN ON X-A                                           00280
      ITITLE(12) = 8HXIS                                                00290
      CALL DRAW(NUMX,X,XSTAT,0,0,4H    ,ITITLE,0.,0.,0,0,0,0,8,8,1,LAST)00300
      ITITLE(11) = 8HN ON Y-A                                           00310
      CALL DRAW(NUMY,Y,YSTAT,0,0,4H    ,ITITLE,0.,0.,0,0,0,0,8,8,1,LAST)00320
      ITITLE(1) = 8HKRUMM AN                                            00330
      ITITLE(2) = 8HD SNYDER                                            00340
```

```
      ITITLE(3) = 8H                                                    00350
      ITITLE(4) = 8H      LE                                            00360
      ITITLE(5) = 8HAST SQUA                                            00370
      ITITLE(6) = 8HRES FIT                                             00380
      ITITLE(7) = 8HTO MINE                                             00390
      ITITLE(8) = 8HCUMULATI                                            00400
      ITITLE(9) = 8HVE DISTR                                            00410
      ITITLE(10) = 8HIBUTION                                            00420
      ITITLE(11) = 8HCURVE ON                                           00430
      ITITLE(12) = 8H X-AXIS                                            00440
      CALL LSCFWOP(X,XSTAT,NUMX,KK,0,1,1,4,0,1,1,0,0,.0000001,0,ITITLE) 00450
      ITITLE(12) = 8H Y-AXIS                                            00460
      CALL LSCFWOP(Y,YSTAT,NUMY,KK,0,1,1,4,0,1,1,0,0,.0000001,0,ITITLE) 00470
      IR=54137                                                          00490
      ITITLE(4)= 8H                                                     00491
      ITITLE(5) = 8HRANDOM M                                            00492
      ITITLE(6) = 8HINEFIELD                                            00493
      ITITLE(7) = 8H    RESU                                            00494
      ITITLE(8) = 8HLTING FR                                            00495
      ITITLE(9) = 8HOM OUTPU                                            00496
      ITITLE(10) = 8HT OF PRO                                           00497
      ITITLE(11) = 8HGRAM MIN                                           00498
      ITITLE(12) = 8HDEL                                                00499
      CALL DRAW(NF,FXPTS,FYPTS,1,0,4H    ,ITITLE,2000.,2000.,0,0,0,0,8,8 00500
     1,0,LAST)                                                          00502
      CALL POSGEN(X,XSTAT,Y,YSTAT,IR,N,NUMX,NUMY,ITITLE)                00505
      END                                                               00510
```

```
      SUBROUTINE POSGEN(A,B,C,D,IR,N,NUMX,NUMY,ITITLE)                    00520
      DIMENSION A(500),B(500),C(500),D(500),XPOSIT(500),YPOSIT(500),      00530
     1ITITLE(12)                                                          00535
      DO 100 J=1,N                                                        00540
      CALL RAN1(IR,RN)                                                    00550
      DO 1 I=1,NUMX                                                       00560
      IF(B(I)-RN)1,2,3                                                    00570
    1 CONTINUE                                                            00580
    2 XPOSIT(J)= A(I)                                                     00590
      GO TO 10                                                            00600
    3 FRAC = (B(I)-RN)/(B(I)-B(I-1))                                      00610
      XPOSIT(J) = A(I-1) + (A(I)-A(I-1))*FRAC                             00620
   10 CALL RAN1(IR,RN)                                                    00630
      DO 4 I=1,NUMY                                                       00635
      IF(D(I)-RN)4,5,6                                                    00640
    4 CONTINUE                                                            00650
    5 YPOSIT(J)=C(I)                                                      00660
      GO TO 100                                                           00670
    6 FRAC = (D(I)-RN)/(D(I)-D(I-1))                                      00680
      YPOSIT(J) = C(I-1) + (C(I)-C(I-1))*FRAC                             00690
  100 CONTINUE                                                            00700
      PRINT 200                                                           00710
  200 FORMAT(1H1,50X,17H RANDOM MINEFIELD // 35X,46H MINE NO.          X  00720
     1 COORD              Y COORD //)                                     00730
      DO 250 I=1,N                                                        00740
      PRINT 300,I,XPOSIT(I),YPOSIT(I)                                     00750
  300 FORMAT(37X,I5,10X,F10.4,10X,F10.4)                                  00760
  250 CONTINUE                                                            00770
      AN = FLOATF(N)                                                      00771
      DIV = AN/30.                                                        00772
      IDIV = XFIXF(DIV)                                                   00773
      NREM = N - 30*IDIV                                                  00774
      IF(NREM.EQ.0)50,60                                                  00775
   60 DO 61 I=1,IDIV                                                      00777
```

```
      CALL DRAW(30,XPOSIT,YPOSIT,2,2,4H    ,ITITLE,2000.,2000.,0,0,0,0,8   00778
     1,8,0,LAST)                                                           00779
      LU = N - 30*I                                                        00780
      DO 62 J=1,LU                                                         00781
      XPOSIT(J) = XPOSIT(J+30)                                             00782
   62 YPOSIT(J) = YPOSIT(J+30)                                             00783
   61 CONTINUE                                                             00784
      IF(NREM.EQ.1)63,64                                                   00785
   63 XPOSIT(2) = XPOSIT(1)  $  YPOSIT(2) = YPOSIT(1)                      00786
      CALL DRAW(2,XPOSIT,YPOSIT,3,2,4H    ,ITITLE,2000.,2000.,0,0,0,0,8,   00787
     18,0,LAST)                                                            00788
      GO TO 9000                                                           00789
   64 CALL DRAW(NREM,XPOSIT,YPOSIT,3,2,4H    ,ITITLE,2000.,2000.,0,0,0,0   00790
     1,8,8,0,LAST)                                                         00791
      GO TO 9000                                                           00792
   50 DO 51 I=1,IDIV                                                       00793
      IF(I.EQ.IDIV)55,56                                                   00794
   56 CALL DRAW(30,XPOSIT,YPOSIT,2,2,4H    ,ITITLE,2000.,2000.,0,0,0,0,8   00795
     1,8,0,LAST)                                                           00796
      LU = N - 30*I                                                        00797
      DO 52 J=1,LU                                                         00798
      XPOSIT(J) = XPOSIT(J+30)                                             00799
   52 YPOSIT(J) = YPOSIT(J+30)                                             00800
   51 CONTINUE                                                             00801
   55 CALL DRAW(30,XPOSIT,YPOSIT,3,2,4H    ,ITITLE,2000.,2000.,0,0,0,0,8   00802
     1,8,0,LAST)                                                           00803
 9000 RETURN                                                               00804
      END                                                                  00805
```

```
      SUBROUTINE RAN1(IR,RN)                                          00806
      A=0.                                                            00810
      B=1.                                                            00820
      IR=3125*IR                                                      00830
      IR=IR-(IR/67108864)*67108864                                    00840
      FY=FLOATF(IR)                                                   00850
      RN=FY/67108864.*(B-A)+A                                         00860
      RETURN                                                          00870
      END                                                             00880
```

```
      SUBROUTINE LSCFWOP(A,B,MM,KK,KOT,KIP,KAP,KOP,NOP,NIP,MTI,XBAR,GAMM00000002
     1A,TOL,MC,ITITLE)                                                000    4
      DIMENSION VEC(3000,6),S(101),ALP(101),BETA(101),RHOSQ(101),XP(101)00000010
      DIMENSION A(500),B(500),ITITLE(12)                               00000020
      COMMON VEC,S,ALP,BETA,XP                                         00000 30
    2 FORMAT(1H1)                                                      00000110
   21 FORMAT(10I5)                                                     00000180
      DO 6 II=1,MM                                                     00000190
      VEC(II,1)=A(II)                                                  00000200
    6 VEC(II,2) = B(II)                                                00000220
      CALL SHIFT(XBAR,GAMMA,MM)                                        00000270
      S1 = MM                                                          00000280
      DO 31 J = 1,MM                                                   00000290
      VEC(J,3) = 0.0                                                   00000300
   31 VEC(J,4) = 1.0                                                   00000310
      BETA(1) = 0.0                                                    00000320
      S3 = 0.0                                                         00000330
      DO 32 L = 1,MM                                                   00000340
   32 S3 = S3 + VEC(L,2)**2                                            00000350
      K2 = KK + 1                                                      00000360
      DO 302 I = 1,K2                                                  00000370
      S4 = 0.0                                                         00000380
      DO 33 J = 1,MM                                                   00000390
   33 S4 = S4 + VEC(J,2)*VEC(J,4)                                      00000400
      S(I) = S4/S1                                                     00000410
      S5 = S3-S(I)**2*S1                                               00000420
      IF(MM-I-1)305,305,301                                            00000430
  301 RHOSQ(I) = S5/FLOATF(MM-I-1)                                     00000440
      GO TO 303                                                        00000450
  305 RHOSQ(I) = S5                                                    00000460
  303 K1 = I -1                                                        00000470
      DO 34 K = 1,K1                                                   00000480
      IF(ABSF(RHOSQ(K)-RHOSQ(I))-TOL) 71,71,34                         00000490
   34 CONTINUE                                                         00000500
      IF(I - K2)35,72,72                                               00000510
```

```
   35 DO 36 N = 1,MM                                                    00000520
   36 VEC(N,5) = VEC(N,1)*VEC(N,4)                                      00000530
      T1 = 0.0                                                          0000 540
      DO 37 JJJ = 1,MM                                                  00000550
   37 T1 = T1 + VEC(JJJ,5)*VEC(JJJ,4)                                   00000560
      ALP(I+1) = T1/S1                                                  00000570
      T2 = 0.0                                                          00000580
      DO 38 M = 1,MM                                                    00000590
      VEC(M,5) = (VEC(M,1)-ALP(I+1))*VEC(M,4)-BETA(I)*VEC(M,3)          00000600
      T2 = T2+VEC(M,5)**2                                               00000610
      VEC(M,3) = VEC(M,4)                                               00000620
   38 VEC(M,4) = VEC(M,5)                                               00000630
      BETA(I+1) = T2/S1                                                 00000640
      S3 = S5                                                           00000650
      S1 = T2                                                           00000660
  302 CONTINUE                                                          00000670
   72 K3 = K2                                                           00000680
      GO TO 40                                                          00000690
   71 K3 = I                                                            00000700
   40 CALL UNSHIFT(XBAR,GAMMA,MM)                                       00000710
      KGL= 2                                                            00000720
      IF(KIP) 718,718,717                                               00000730
  717 CALL POLK(K3,KAP)                                                 00000740
  718 IF(NOP) 420,719,719                                               00000750
  719 DO 7000 J=1,MM                                                    00000760
 7000 VEC(J,5) = VEC(J,2)                                               00000770
      IF(NOP)420,405,407                                                00000780
  407 PRINT 1001                                                        00000790
 1001 FORMAT(1H2/50H THE ORDINATE VALUES USING THE RECURRENCE FORMULAS) 00000800
      PRINT 2                                                           00000810
      MC = MM                                                           00000820
      GO TO 1002                                                        00000830
  405 CALL SHIFT(XBAR,GAMMA,MM)                                         00000840
      DO 406 N =1,MM                                                    00000850
      VEC(N,2) = XP(1)                                                  00000860
```

```
      K4 = K3 - 1                                                        00000870
      DO 406 K=1,K4                                                      00000880
  406 VEC(N,2) = VEC(N,2)*VEC(N,1) + XP(K+1)                             00000890
      CALL UNSHIFT(XBAR,GAMMA,MM)                                        00000900
      DO 7001 J=1,MM                                                     00000903
 7001 B(J) = VEC(J,2)                                                    00000905
      CALL DRAW(MM,A,B,0,0,4H    ,ITITLE,0.,0.,0,0,0,0,8,8,1,LAST)       00000907
      IF(MTI-K4) 4760,4760,420                                           00000910
 4760 PRINT 5,K4                                                         00000920
    5 FORMAT(1H2/73H THE ORDINATE VALUES AS EVALUATED BY GENERATING THE  00000930
     1POLYNOMIAL OF DEGREE I4/1H1)                                       00000940
      CALL XPRINT(MM,KOP)                                                00000950
C                                                                        0000 955
C STATEMENT NO. 420 USED TO READ IF(KOT)1327,1327,402                    00000960
C                                                                        0000 965
  420 IF(KOT)411,411,402                                                 00000967
  402 READ 21,MC                                                         00000970
      PRINT 821                                                          00000980
  821 FORMAT(1H2/61H THE FOLLOWING ORDINATES ARE CALCULATED FOR THE NEW  00000990
     1ABSCISSAS)                                                         00001000
      KGL = 1                                                            00001010
      KOT = KOT - 1                                                      00001020
      READ   7  ,(VEC(J,1),J=1,MC)                                       00001025
    7 FORMAT(6F10.5)                                                     00001030
 1002 K2 = K3 - 1                                                        00001040
      NOP = 0                                                            00001050
      CALL SHIFT(XBAR,GAMMA,MC)                                          00001060
      DO 51 J = 1,MC                                                     00001070
      VEC(J,2) = 0.0                                                     00001080
      VEC(J,3) = 0.0                                                     00001090
   51 VEC(J,4) = 1.0                                                     00001100
      DO 502 I = 1,K3                                                    00001110
      DO 52 J = 1,MC                                                     00001120
      VEC(J,2) = VEC(J,2) + S(I)*VEC(J,4)                                00001130
      IF(I -K3)503,52,52                                                 00001140
```

```
  503  P2 = (VEC(J,1)-ALP(I+1))*VEC(J,4) - BETA(I)*VEC(J,3)                    00001150
       VEC(J,3) = VEC(J,4)                                                     00001160
       VEC(J,4) = P2                                                           00001170
    52 CONTINUE                                                               00001180
       CALL UNSHIFT(XBAR,GAMMA,MC)                                             00001190
       JJ2 = I - 1                                                             00001200
       IF(NIP)1003,1003,1004                                                   00001210
 1004  IF(I-K3)502,1003,502                                                    00001220
 1003  IF(MTI-JJ2) 5436,5436,502                                               00001230
 5436  PRINT 643,JJ2                                                           00001240
  643  FORMAT(////66H THE FOLLOWING VALUES WERE CALCULATED USING A POLYNO00001250
      1MIAL OF DEGREE I4)                                                     00001260
       GO TO (705,799), KGL                                                    00001270
  799  CALL XPRINT(MC,KOP)                                                     00001280
       GO TO 502                                                               00001290
  705  PRINT 707                                                               00001300
  707  FORMAT(/3(8X,29H ABSCISSA          ORDINATE   ))                        00001310
       MC1 = MC - 2                                                            00001320
       DO 701 K = 1,MC1,3                                                      00001330
       L = K                                                                   00001340
  701  PRINT 23,VEC(K,1),VEC(K,2),VEC(K+1,1),VEC(K+1,2),VEC(K+2,1),           00001350
      1  VEC(K+2,2)                                                           00001360
       IF(L+1-MC1)703,702,502                                                  00001370
  702  PRINT 23,VEC(MC,1),VEC(MC,2)                                            00001380
       GO TO 502                                                               00001390
  703  PRINT 23,VEC(MC-1,1),VEC(MC-1,2),VEC(MC,1),VEC(MC,2)                    00001400
  502  CALL SHIFT(XBAR,GAMMA,MC)                                               00001410
 5002  CONTINUE                                                                00001420
   23  FORMAT(3(5X,1PE14.4,3X,1PE14.4))                                        00001430
C                                                                              00001435
C  THE NEXT STATEMENT USED TO READ IF(KOT)1327,1327,402                        00001440
C                                                                              00001445
   411 CONTINUE                                                                00001450
       END                                                                     00001460
```

```
      SUBROUTINE POLK(KQ,KR)                                            00001470
      DIMENSION COEF(100,100),ALP(101),BETA(101),S(101),XP(101),        00001480
     1 DUM(8000)                                                        00001490
      COMMON DUM,COEF,S,ALP,BETA,XP                                     00001500
      DO 31 I = 1,KQ                                                    00001510
      COEF(I,1) = 1.0                                                   00001520
   31 COEF(I,I+1) = 0.0                                                 00001530
      DO 32 I = 2,KQ                                                    00001540
   32 COEF(I,2) = COEF(I-1,2) - ALP(I)                                  00001550
      DO 33 J = 3,KQ                                                    00001560
      DO 33 I = J,KQ                                                    00001570
   33 COEF(I,J) = COEF(I-1,J) - ALP(I)*COEF(I-1,J-1)  -                 00001580
     1  BETA(I-1)*COEF(I-2,J-2)                                         00001590
      DO 34 I = 1,KQ                                                    00001600
      DO 34 J = 1,I                                                     00001610
   34 COEF(I,J) = COEF(I,J)*S(I)                                        00001620
   17 FORMAT(///46H THE COEFFICIENTS FOR THE POLYNOMIAL OF DEGREE I4//(/ 00001630
     1 1PE20.10,1PE20.10,1PE20.10,1PE20.10,1PE20.10,1PE20.10))          00001640
      DO 26 N = 1,KQ                                                    00001650
      MKM = N - 1                                                       00001660
      XP(N) = 0.0                                                       00001670
      DO 531 I=1,N                                                      00001680
  531 XP(N) = COEF(I,I) + XP(N)                                         00001690
      IF(MKM) 400,350,400                                               00001700
  400 DO 35 J=1,MKM                                                     00001710
      XP(N-J) = 0.0                                                     00001720
      M = J + 1                                                         00001730
      DO 35 I = M,N                                                     00001740
   35 XP(N-J) = COEF(I,I-J) + XP(N-J)                                   00001750
  350 IF(KR) 21,26,23                                                   00001760
   21 MN1 = N - 1                                                       00001770
      PRINT 17,MN1,(XP(L),L=1,N)                                        00001780
      GO TO 26                                                          00001790
   23 IF(N-KQ)26,21,26                                                  00001800
   26 CONTINUE                                                          00001810
      END                                                               00001820
```

```
      SUBROUTINE SHIFT(X,Y,MQ)                                          00001830
      DIMENSION VEC(3000,6),S(101),ALP(101),BETA(101),XP(101)           00001840
      COMMON VEC,S,ALP,BETA,XP                                          00001850
      DO 31 I=1,MQ                                                      00001860
   31 VEC(I,1)=(VEC(I,1)-X)/(1.0-Y)                                     00001870
      END                                                               00001880
```

```
      SUBROUTINE UNSHIFT(X,Y,MQ)                                        00001890
      DIMENSION VEC(3000,6),S(101),ALP(101),BETA(101),XP(101)           00001900
      COMMON VEC,S,ALP,BETA,XP                                          00001910
      DO 31 I=1,MQ                                                      00001920
   31 VEC(I,1)=VEC(I,1)*(1.0-Y) + X                                     00001930
      END                                                               00001940
```

```
      SUBROUTINE XPRINT(NUM,KOP)                                        00001950
      DIMENSION VEC(3000,6),S(101),ALP(101),BETA(101),XP(101)           00001960
      COMMON VEC,S,ALP,BETA,XP                                          00001970
      YBAR = 0.0                                                        00001980
      YAV  = 0.0                                                        00001990
      YSQ = 0.0                                                         00002000
      YMAX = 0.0                                                        00002010
      DO 950 J=1,NUM                                                    00002020
      VEC(J,6)= VEC(J,2)-VEC(J,5)                                       00002030
      T = ABSF(VEC(J,6))                                                00002040
      YBAR = YBAR + T                                                   00002050
      YSQ = YSQ + T*T                                                   00002060
      IF(YMAX-T) 970,950,950                                            00002070
  970 YMAX = T                                                          00002080
  950 YAV = YAV + VEC(J,6)                                              00002090
      YBAR = YBAR/FLOATF(NUM)                                           00002100
      YSQ = SQRTF(YSQ/FLOATF(NUM))                                      00002110
      GO TO (900,902,904,905),KOP                                       00002120
  904 PRINT 966                                                         00002130
  966 FORMAT(////9X,8HORIGINAL,11X,8HCOMPUTED,11X,8HORDINATE/9X,8HABSCIS00002140
     1SA,11X,8HORDINATE,10X,11HDIFFERENCES//)                           00002150
      PRINT 965,(VEC(J,1),VEC(J,2),VEC(J,6),J=1,NUM)                    00002160
  965 FORMAT(3(5X,1PE14.4))                                             00002170
      GO TO 900                                                         00002180
  902 PRINT 964                                                         00002190
  964 FORMAT(////9X,8HORIGINAL,11X,8HORIGINAL,11X,8HORDINATE/9X,8HABSCIS00002200
     1SA,11X,8HORDINATE,10X,11HDIFFERENCES//)                           00002210
      PRINT 965,(VEC(J,1),VEC(J,5),VEC(J,6),J=1,NUM)                    00002220
      GO TO 900                                                         00002230
  905 PRINT 962                                                         00002240
  962 FORMAT(////9X,8HORIGINAL,11X,8HORIGINAL,11X,8HCOMPUTED,11X,8HORDIN00002250
     1ATE/9X,8HABSCISSA,11X,8HORDINATE,11X,8HORDINATE,10X,11HDIFFERENCES00002260
     2//)                                                               00002270
      PRINT 961,(VEC(J,1),VEC(J,5),VEC(J,2),VEC(J,6),J=1,NUM)           00002280
  961 FORMAT(4(5X,1PE14.4))                                             00002290
```

```
      900 PRINT 960, YAV, YBAR , YSQ , YMAX                                    00002300
      960 FORMAT(////53H SUM OF (COMPUTED ORDINATE MINUS ORIGINAL ORDINATE)=00002310
         1 1PE16.8//17H ERROR NORM L1 = 1PE16.8//17H ERROR NORM L2 = 1PE16.800002320
         2 //25H ERROR NORM L-INFINITY = 1PE16.8)                              00002330
          GO TO (1900,2000,2000,2000),KOP                                      00002340
     1900 PRINT 4                                                              00002350
        4 FORMAT(1H2)                                                          00002360
          RETURN                                                               00002370
     2000 PRINT 5                                                              00002380
        5 FORMAT(1H1)                                                          00002390
            END                                                                0000240
```

# APPENDIX XIII

## SAMPLE RUN

On the succeeding pages of this appendix appears the printed output of programs MINDEL and MINDEL1. The particular runs shown were chosen on the basis of the number of program features which were demonstrated,

SUMMARY NARRATIVE

| TIME | EVENT |
|---|---|
| 160850. | SUBMARINE NO. 1 ARRIVES AT IP FOR LEG NO. 1 |
| 160842. | MINE NO. 149 LAYED |
| 160849. | MINE NO. 150 NOT LAYED DUE TO EXCESSIVE TIME DELAY |
| 160851. | MINE NO. 151 LAYED |
| 160901. | MINE NO. 152 LAYED |
| 160910. | MINE NO. 153 LAYED |
| 160920. | MINE NO. 154 LAYED |
| 160935. | SUB NO. 1 ARRIVES AT TP FOR LEG NO. 1 |
| 160959. | SUB NO. 1 ARRIVES AT IP FOR LEG NO. 2 |
| 161000. | MINE NO. 155 LAYED |
| 161004. | MINE NO. 156 LAYED |
| 161006. | MINE NO. 157 LAYED |
| 161009. | MINE NO. 158 LAYED |
| 161011. | MINE NO. 159 LAYED |
| 161018. | SUB NO. 1 ARRIVES AT TP FOR LEG NO. 2 |
| 180936. | SUBMARINE NO. 2 ARRIVES AT IP FOR LEG NO. 3 |
| 180942. | MINE NO. 160 LAYED |
| 180945. | MINE NO. 161 LAYED |
| 180948. | MINE NO. 162 LAYED |
| 180951. | MINE NO. 163 LAYED |
| 180953. | MINE NO. 164 LAYED |
| 181000. | SUB NO. 2 SUSPENDS OPS DUE TO ERRONEOUS KNOWLDEGE OF DETECTION. |
| 200244. | SUB NO. 2 RESUMES OPS. |
| 200302. | SUB NO. 2 ARRIVES AT IP FOR LEG NO. 4 |
| 200305. | MINE NO. 165 LAYED |
| 200308. | MINE NO. 166 LAYED |
| 200311. | MINE NO. 167 LAYED |
| 200315. | SUB NO. 2 ARRIVES AT TP FOR LEG NO. 4 |
| 200730. | SHIP NO. 1 ARRIVES AT IP FOR LEG NO. 1 |

200731.5 MINE NO. 64 LAYED
200732.6 MINE NO. 65 LAYED
200733.6 MINE NO. 66 LAYED
200734.7 MINE NO. 67 LAYED
200735.7 MINE NO. 68 LAYED
200736.6 MINE NO. 69 LAYED
200737.6 MINE NO. 70 LAYED
200738.7 MINE NO. 71 LAYED
200739.6 MINE NO. 72 LAYED
200740.6 MINE NO. 73 LAYED
200741.9 MINE NO. 74 LAYED
200743.0 MINE NO. 75 LAYED
200743.9 MINE NO. 76 LAYED
200745.0 MINE NO. 77 LAYED
200746.0 MINE NO. 78 LAYED
200746.9 SHIP NO. 1 DAMAGED. NAVIGATIONAL ERROR DOUBLED.
200748.7 MINE NO. 79 LAYED
200750.8 MINE NO. 80 LAYED
200751.8 MINE NO. 81 LAYED
200752.9 MINE NO. 82 LAYED
200753.8 MINE NO. 83 LAYED
200754.8 MINE NO. 84 LAYED
200755.6 SHIP NO. 1 ARRIVES AT TP FOR LEG NO. 1

200756.9 SHIP NO. 1 ARRIVES AT IP FOR LEG NO. 2
200759.3 MINE NO. 85 LAYED
200801.9 MINE NO. 86 LAYED
200804.6 MINE NO. 87 LAYED
200807.2 MINE NO. 88 LAYED
200810.1 MINE NO. 89 LAYED
200812.6 MINE NO. 90 LAYED
200815.5 MINE NO. 91 LAYED
200818.2 MINE NO. 92 LAYED
200818.9 SHIP NO. 1 ARRIVES AT TP FOR LEG NO. 2

200745.3 SHIP NO. 2 ARRIVES AT IP FOR LEG NO. 3
200747.1 SHIP NO. 2 DAMAGED. ABORTS MISSION. MINES REMAINING ON BOARD ARE 93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109

```
                                                                    110
                                                                    111
                                                                    112
                                                                    113
200900.0 SHIP NO.   3 ARRIVES AT IP FOR LEG NO.   5
200902.6 MINE NO. 114 LAYED
200905.1 MINE NO. 115 LAYED
200907.5 MINE NO. 116 LAYED
200910.3 MINE NO. 117 LAYED
200912.8 MINE NO. 118 LAYED
200915.4 MINE NO. 119 LAYED
200917.9 MINE NO. 120 LAYED
200920.3 MINE NO. 121 LAYED
200923.1 MINE NO. 122 LAYED
200925.5 MINE NO. 123 LAYED
200928.2 MINE NO. 124 LAYED
200930.5 SHIP NO.   3 DAMAGED. NAVIGATIONAL ERROR DOUBLED.
200930.6 MINE NO. 125 LAYED
200933.1 MINE NO. 126 LAYED
200935.1 SHIP NO.   3 ARRIVES AT TP FOR LEG NO.   5

200936.7 SHIP NO.   3 ARRIVES AT IP FOR LEG NO.   6
200939.4 MINE NO. 124 LAYED
200943.0 MINE NO. 128 LAYED
200946.8 MINE NO. 129 LAYED
200950.1 MINE NO. 130 LAYED
200954.0 MINE NO. 131 LAYED
200957.7 MINE NO. 132 LAYED
201000.9 MINE NO. 133 LAYED
201004.4 MINE NO. 134 LAYED
201007.8 MINE NO. 135 LAYED
201009.9 MINE NO. 136 LAYED
201012.9 SHIP NO.   3 ARRIVES AT TP FOR LEG NO.   6

201015.6 SHIP NO.   3 ARRIVES AT IP FOR LEG NO.   7
201019.1 MINE NO. 137 LAYED
201021.8 MINE NO. 138 LAYED
201024.8 MINE NO. 139 LAYED
201027.9 MINE NO. 140 LAYED
201030.6 MINE NO. 141 LAYED
201033.7 MINE NO. 142 LAYED
201036.9 MINE NO. 143 LAYED
201039.7 MINE NO. 144 LAYED
201042.9 MINE NO. 145 LAYED
201045.9 MINE NO. 146 LAYED
201049.2 MINE NO. 147 LAYED
201050.5 MINE NO. 148 LAYED
```

201052.2 SHIP NO. 5 ARRIVES AT IP FOR LEG NO. 7

221550.5 A/C NO. 1 ARRIVES AT IP FOR LEG NO. 1
221551.4 MINE NO. 1 LAYED
221551.5 MINE NO. 2 LAYED
221551.6 MINE NO. 3 LAYED
221551.7 MINE NO. 4 LAYED
221551.8 MINE NO. 5 LAYED
221551.9 MINE NO. 6 LAYED
221552.0 MINE NO. 7 LAYED
221552.1 MINE NO. 8 LAYED
221553.7 A/C NO. 1 ARRIVES AT TP FOR LEG NO. 1

221554.1 A/C NO. 2 ARRIVES AT IP FOR LEG NO. 2
221555.0 MINE NO. 9 LAYED
221555.1 MINE NO. 10 LAYED
221555.2 MINE NO. 11 LAYED
221555.3 MINE NO. 12 LAYED
221555.4 MINE NO. 13 LAYED
221555.5 MINE NO. 14 LAYED
221555.6 MINE NO. 15 LAYED
221555.7 MINE NO. 16 LAYED
221557.3 A/C NO. 2 ARRIVES AT TP FOR LEG NO. 2

221557.8 A/C NO. 3 ARRIVES AT IP FOR LEG NO. 3
221558.7 MINE NO. 17 LAYED
221558.8 MINE NO. 18 LAYED
221558.9 MINE NO. 19 LAYED
221559.0 MINE NO. 20 LAYED
221559.1 MINE NO. 21 LAYED
221559.2 MINE NO. 22 LAYED
221559.3 MINE NO. 23 LAYED
221559.7 MINE NO. 24 LAYED
221601.0 A/C NO. 3 ARRIVES AT TP FOR LEG NO. 3

221603.0 A/C NO. 4 ARRIVES AT IP FOR LEG NO. 4
221603.7 A/C NO. 4 ATTRITED. MINES REMAINING ON BOARD ARE
25
29
26
30
27
31
28
32

221607.3 A/C NO. 5 ARRIVES AT IP FOR LEG NO. 5
221608.2 MINE NO. 33 DID NOT RELEASE
221608.3 MINE NO. 34 DID NOT RELEASE
221608.4 MINE NO. 35 LAYED
221608.5 MINE NO. 36 LAYED
221608.6 MINE NO. 37 IS BLOCKED BY PRIOR RELEASE FAILURE

```
221608.7 MINE NO.  38 IS BLOCKED BY PRIOR RELEASE FAILURE
221608.8 MINE NO.  39 DID NOT RELEASE
221608.9 MINE NO.  40 DID NOT RELEASE
221610.5 A/C NO.   5 ARRIVES AT TP FOR LEG NO.   5

221611.9 A/C NO.   6 ARRIVES AT IP FOR LEG NO.   6
221612.8 MINE NO.  41 LAYED
221612.9 MINE NO.  42 DID NOT RELEASE
221613.0 MINE NO.  43 LAYED
221613.1 MINE NO.  44 LAYED
221613.2 MINE NO.  45 LAYED
221613.3 MINE NO.  46 IS BLOCKED BY PRIOR RELEASE FAILURE
221613.4 MINE NO.  47 LAYED
221613.5 MINE NO.  48 DID NOT RELEASE
221615.1 A/C NO.   6 ARRIVES AT TP FOR LEG NO.   6

221612.2 A/C NO.   7 ARRIVES AT IP FOR LEG NO.   7
221613.0 MINE NO.  49 LAYED
221613.1 MINE NO.  50 LAYED
221613.1 MINE NO.  51 DID NOT RELEASE
221613.2 MINE NO.  52 DID NOT RELEASE
221613.3 MINE NO.  53 LAYED
221613.4 MINE NO.  54 LAYED
221613.5 MINE NO.  55 LAYED
221614.8 A/C NO.   7 ARRIVES AT TP FOR LEG NO.   7

221617.3 A/C NO.   7 ARRIVES AT IP FOR LEG NO.   8
221618.1 MINE NO.  56 LAYED
221618.1 MINE NO.  57 IS BLOCKED BY PRIOR RELEASE FAILURE
221618.2 MINE NO.  58 IS BLOCKED BY PRIOR RELEASE FAILURE
221618.3 MINE NO.  59 LAYED
221618.4 MINE NO.  60 LAYED
221618.5 MINE NO.  61 LAYED
221618.6 MINE NO.  62 LAYED
221618.7 MINE NO.  63 IS BLOCKED BY PRIOR RELEASE FAILURE
221620.0 A/C NO.   7 ARRIVES AT TP FOR LEG NO.   8
```

## INPUT TO MINE DELIVERY PHASE

### FIELD DATA

THE X REFERENCE POSITION IS -121.9745 DEGREES
THE Y REFERENCE POSITION IS 36.6287 DEGREES

FIELD PERIMETER POINTS

| XCOORD | YCOORD |
|---|---|
| 1743.5314 | 829.2722 |
| 1743.5314 | 4045.2309 |
| 9825.6956 | 6917.3448 |
| 15626.6038 | 6917.3448 |
| 14648.9228 | 3842.9694 |
| 10281.9468 | 3842.9694 |
| 11357.3960 | 1031.5338 |
| 8489.5313 | 0 |
| 7430.3767 | 2851.8879 |
| 1743.5314 | 829.2722 |



### VEHICLE ARRIVAL LIST

| TOA | VEHICLE TYPE | VEHICLE NUMBER | TOTAL NO. OF LEGS ASSIGNED | NO. OF BOMB RACKS | LEGS ASSIGNED (CODED) | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| 160830.0 | 2 | 1 | 2 | 0 | 1002000 | 0 | 0 | 0 | 0 | 0 |
| 180936.0 | 1 | 2 | 2 | 0 | 3004000 | 0 | 0 | 0 | 0 | 0 |
| 200730.5 | 10 | 1 | 2 | 0 | 1002000 | 0 | 0 | 0 | 0 | 0 |
| 200745.3 | 10 | 2 | 2 | 0 | 3004000 | 0 | 0 | 0 | 0 | 0 |
| 200900.0 | 11 | 3 | 3 | 0 | 5006007 | 0 | 0 | 0 | 0 | 0 |
| 221550.5 | 100 | 1 | 1 | 4 | 1000000 | 0 | 0 | 0 | 0 | 0 |
| 221554.1 | 100 | 2 | 1 | 4 | 2000000 | 0 | 0 | 0 | 0 | 0 |
| 221557.8 | 100 | 3 | 1 | 4 | 3000000 | 0 | 0 | 0 | 0 | 0 |
| 221603.0 | 101 | 4 | 1 | 4 | 4000000 | 0 | 0 | 0 | 0 | 0 |
| 221607.3 | 101 | 5 | 1 | 4 | 5000000 | 0 | 0 | 0 | 0 | 0 |
| 221611.9 | 101 | 6 | 1 | 4 | 6000000 | 0 | 0 | 0 | 0 | 0 |
| 221612.2 | 102 | 7 | 2 | 6 | 7008000 | 0 | 0 | 0 | 0 | 0 |

### AIRCRAFT CHARACTERISTICS LIST

| VEHICLE TYPE | XSIT SPEED | M/L SPEED | M/L ALTITUDE | PROB OF MINE RELEASE | PROB OF SURVIVAL OVER OBJECTIVE AREA |
|---|---|---|---|---|---|
| 100 | 225 | 200 | 1000 | .88 | .80 |
| 101 | 250 | 200 | 1000 | .90 | .76 |
| 102 | 400 | 250 | 5000 | .94 | .85 |

AIRCRAFT LEG DESCRIPTION LIST

| LEG NO. | NO.OF MINES THIS LEG | AIRCRAFT INITIAL POSIT XCOORD | YCOORD | MINE NO. | MINE INTENDED POSITION XCOORD | YCOORD | AIRCRAFT TERMINAL POINT XCOORD | YCOORD |
|---|---|---|---|---|---|---|---|---|
| 1 | 8 | 21671.933 | 6755.536 | 1 | 15610.309 | 6755.536 | | |
| | | | | 2 | 14942.227 | 6755.536 | | |
| | | | | 3 | 14241.556 | 6755.536 | | |
| | | | | 4 | 13492.000 | 6755.536 | | |
| | | | | 5 | 12840.213 | 6755.536 | | |
| | | | | 6 | 12139.541 | 6755.536 | | |
| | | | | 7 | 11488.870 | 6755.536 | | |
| | | | | 8 | 10721.903 | 6755.536 | | |
| | | | | | | | 195.536 | 9526.519 |
| 2 | 8 | 21313.449 | 6411.691 | | | | | |
| | | | | 9 | 15251.826 | 6411.691 | | |
| | | | | 10 | 14388.208 | 6411.691 | | |
| | | | | 11 | 13806.778 | 6411.691 | | |
| | | | | 12 | 13231.285 | 6411.691 | | |
| | | | | 13 | 12530.614 | 6411.691 | | |
| | | | | 14 | 11895.121 | 6411.691 | | |
| | | | | 15 | 11096.681 | 6411.691 | | |
| | | | | 16 | 10412.304 | 6411.691 | | |
| | | | | | | | 195.536 | 9526.519 |
| 3 | 8 | 21460.192 | 6027.394 | | | | | |
| | | | | 17 | 15333.300 | 6027.394 | | |
| | | | | 18 | 14632.628 | 6027.394 | | |
| | | | | 19 | 13948.251 | 6027.394 | | |
| | | | | 20 | 13296.464 | 6027.394 | | |
| | | | | 21 | 12612.087 | 6027.394 | | |
| | | | | 22 | 11911.415 | 6027.394 | | |
| | | | | 23 | 11243.333 | 6027.394 | | |
| | | | | 24 | 8945.783 | 6027.394 | | |
| | | | | | | | 195.536 | 9526.519 |
| 4 | 8 | 21124.208 | 5582.419 | | | | | |
| | | | | 25 | 15007.406 | 5582.419 | | |
| | | | | 26 | 14323.029 | 5582.419 | | |
| | | | | 27 | 13638.652 | 5582.419 | | |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | | 28 | 12937.981 | 5582.419 | | |
| | | | | 29 | 12221.014 | 5582.419 | | |
| | | | | 30 | 11569.227 | 5582.419 | | |
| | | | | 31 | 10868.556 | 5582.419 | | |
| | | | | 32 | 10200.473 | 5582.419 | | |
| | | | | | | | 195.536 | 9526.519 |
| | 5 | | 21436.440 | 5198.122 | 33 | 15105.174 | 5198.122 | |
| | | | | | 34 | 14437.092 | 5198.122 | |
| | | | | | 35 | 13687.536 | 5198.122 | |
| | | | | | 36 | 13019.454 | 5198.122 | |
| | | | | | 37 | 12335.077 | 5198.122 | |
| | | | | | 38 | 11666.995 | 5198.122 | |
| | | | | | 39 | 10998.913 | 5198.122 | |
| | | | | | 40 | 10265.652 | 5198.122 | |
| | | | | | | | 195.536 | 9526.519 |
| | 6 | 8 | 20840.903 | 4793.599 | 41 | 14746.691 | 4793.599 | |
| | | | | | 42 | 14078.609 | 4793.599 | |
| | | | | | 43 | 13361.643 | 4793.599 | |
| | | | | | 44 | 12660.971 | 4793.599 | |
| | | | | | 45 | 12009.184 | 4793.599 | |
| | | | | | 46 | 11324.807 | 4793.599 | |
| | | | | | 47 | 10591.546 | 4793.599 | |
| | | | | | 48 | 9907.169 | 4793.599 | |
| | | | | | | | 195.536 | 9526.519 |
| | 7 | 8 | 20838.670 | 4409.302 | 49 | 14359.324 | 4409.302 | |
| | | | | | 50 | 13671.241 | 4409.302 | |
| | | | | | 51 | 12986.865 | 4409.302 | |
| | | | | | 52 | 12286.193 | 4409.302 | |
| | | | | | 53 | 11585.522 | 4409.302 | |
| | | | | | 54 | 10901.145 | 4409.302 | |
| | | | | | 55 | 10200.473 | 4409.302 | |
| | | | | | | | 195.536 | 9526.519 |
| | 8 | 8 | 20808.314 | 3984.553 | 56 | 14665.217 | 3984.553 | |
| | | | | | 57 | 13964.546 | 3984.553 | |
| | | | | | 58 | 13263.874 | 3984.553 | |
| | | | | | 59 | 12530.614 | 3984.553 | |
| | | | | | 60 | 11829.942 | 3984.553 | |
| | | | | | 61 | 11145.565 | 3984.553 | |
| | | | | | 62 | 10428.599 | 3984.553 | |
| | | | | | 63 | 9679.044 | 3984.553 | |

195.536 9526.519

BOMB RACK ARRAY

| A/C NO. | BOMB RACK NO. ONE | TWO | THREE | FOUR | FIVE | SIX |
|---|---|---|---|---|---|---|
| 1 | 5001 | 6002 | 7003 | 8004 | 0 | 0 |
| 2 | 13009 | 14011 | 15011 | 16012 | 0 | 0 |
| 3 | 21017 | 22019 | 23019 | 24020 | 0 | 0 |
| 4 | 29025 | 30026 | 31027 | 32028 | 0 | 0 |
| 5 | 37033 | 38034 | 39035 | 40036 | 0 | 0 |
| 6 | 45041 | 46042 | 47043 | 48044 | 59053 | 60054 |
| 7 | 61055049 | 62056050 | 63057051 | 58052 | | |

## AIRCRAFT MINE CHARACTERISTICS LIST

| MINE TYPE (II) | A | B | C | D | E | F | G | H | I | J | K | L | M |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | .90 | 700 | 800 | .1 | .1 | .6 | .6 | .5 | 3 | 0 | 0 | 0 | .9 |
| 2 | .83 | 650 | 800 | .1 | .1 | .6 | .7 | .5 | 3 | 0 | 0 | 0 | .9 |
| 3 | .81 | 750 | 900 | .1 | .2 | .6 | .8 | .4 | 3 | 0 | 0 | 0 | .9 |
| 4 | .94 | 800 | 950 | .1 | .1 | .6 | .6 | .5 | 3 | 0 | 0 | 0 | .9 |
| 5 | .92 | 640 | 790 | .1 | .2 | .6 | .6 | .5 | 3 | 0 | 0 | 0 | .9 |
| 6 | .85 | 700 | 850 | .2 | .2 | .6 | .8 | .6 | 3 | 0 | 0 | 0 | .9 |
| 7 | .69 | 750 | 900 | .1 | .2 | .6 | .7 | .5 | 3 | 0 | 0 | 0 | .9 |
| 8 | .90 | 750 | 900 | .1 | .2 | .6 | .7 | .5 | 3 | 0 | 0 | 0 | .9 |
| 9 | .92 | 650 | 850 | .1 | .1 | .6 | .6 | .4 | 3 | 0 | 0 | 0 | .9 |
| 10 | 1.00 | 750 | 750 | .1 | .1 | .4 | .4 | .4 | 3 | 0 | 0 | 0 | .9 |
| 11 | .93 | 800 | 950 | .1 | .1 | .6 | .6 | .6 | 3 | 0 | 0 | 0 | .9 |
| 12 | .94 | 750 | 900 | .1 | .1 | .6 | .7 | .5 | 3 | 0 | 0 | 0 | .9 |
| 13 | .83 | 750 | 850 | .1 | .1 | .6 | .7 | .5 | 2 | 0 | .9 | .9 | .9 |
| 14 | .90 | 600 | 750 | .1 | .2 | .6 | .6 | .6 | 3 | 0 | 0 | 0 | .9 |
| 15 | .97 | 650 | 800 | .1 | .1 | .6 | .6 | .5 | 3 | 0 | 0 | 0 | .9 |
| 16 | .91 | 650 | 800 | .1 | .2 | .5 | .6 | .5 | 3 | 0 | 0 | 0 | .9 |
| 17 | .88 | 600 | 750 | .1 | .2 | .5 | .6 | .4 | 3 | 0 | 0 | 0 | .9 |
| 18 | .96 | 750 | 900 | .1 | .2 | .6 | .7 | .4 | 3 | 0 | 0 | 0 | .9 |
| 19 | .92 | 800 | 950 | .1 | .2 | .6 | .7 | .6 | 3 | 0 | 0 | 0 | .9 |
| 20 | .95 | 800 | 950 | .1 | .2 | .6 | .6 | .5 | 3 | 0 | 0 | 0 | .9 |
| 21 | .91 | 750 | 900 | .1 | .2 | .6 | .7 | .5 | 3 | 0 | 0 | 0 | .9 |
| 22 | .94 | 750 | 900 | .1 | .2 | .6 | .7 | .5 | 3 | 0 | 0 | 0 | .9 |
| 23 | .88 | 650 | 800 | .2 | .2 | .6 | .8 | .5 | 3 | 0 | 0 | 0 | .9 |
| 24 | .89 | 700 | 850 | .1 | .2 | .6 | .7 | .4 | 3 | 0 | 0 | 0 | .9 |

A = PROB OF CHUTE DEPLOYMENT
B = CEP FOR 5K FT,CHUTE DEPLOYS
C = CEP FOR 3K FT,CHUTE DOES NOT DEPLOY
D = PROB OF WATER IMPACT DAMAGE,CHUTE DEPLOYS
E = PROB OF WATER IMPACT DAMAGE,CHUTE DOES NOT DEPLOY
F = PROB OF IMPACT DAMAGE ON THE HARDEST BOTTOM,GIVEN DEPTH LESS THAN 20FT AND CHUTE DEPLOYS
G = PROB OF IMPACT DAMAGE ON HARDEST BOTTOM,GIVEN DEPTH LESS THAN 20FT AND CHUTE DOES NOT DEPLOY
H = PROB OF BURYING IN SOFTEST BOTTOM
I = INDICATOR OF MOOR OR BOTTOM TYPE
J = NWL TYPE IF BOTTOM MINE DOES NOT BURY/BURIES
K = PROB OF CASE/ANCHOR SEPARATION
L = PROB OF FULL ANCHOR CABLE DEPLOYMENT
M = PROB MINE OPERATES,GIVEN IT HAS NOT BEEN DAMAGED (RELIABILITY)

SHIP CHARACTERISTIC LIST

| VEHICLE TYPE | XSIT SPEED | M/L SPEED | NAVIGATIONAL ERROR | | | PROB OF MINE RELEASE | MAX TIME DELAY | PROB OF DAMAGE FOR FIRST HALF HOUR IN MLOA |
|---|---|---|---|---|---|---|---|---|
| | | | LEAST | AVERAGE | GREATEST | | | |
| 10 | 25 | 12 | 25 | 100 | 200 | .9 | .2 | .3 |
| 11 | 20 | 8 | 15 | 75 | 150 | .9 | .3 | .4 |

SHIP LEG DESCRIPTION LIST

| LEG NO. | NO. OF MINES THIS LEG | SHIP INITIAL POSIT XCOORD | SHIP INITIAL POSIT YCOORD | MINE NO. | MINE INTENDED POSITION XCOORD | MINE INTENDED POSITION YCOORD | SHIP TERMINAL POINT XCOORD | SHIP TERMINAL POINT YCOORD |
|---|---|---|---|---|---|---|---|---|
| 1 | 21 | 1743.531 | 970.855 | | | | | |
| | | | | 64 | 2134.604 | 1112.439 | | |
| | | | | 65 | 2525.676 | 1334.926 | | |
| | | | | 66 | 2916.749 | 1395.605 | | |
| | | | | 67 | 3307.821 | 1577.640 | | |
| | | | | 68 | 3698.894 | 1678.771 | | |
| | | | | 69 | 4057.377 | 1820.354 | | |
| | | | | 70 | 4415.860 | 1982.163 | | |
| | | | | 71 | 4823.227 | 2164.199 | | |
| | | | | 72 | 5165.416 | 2265.329 | | |
| | | | | 73 | 5556.488 | 2427.139 | | |
| | | | | 74 | 6045.329 | 2548.496 | | |
| | | | | 75 | 6468.990 | 2690.079 | | |
| | | | | 76 | 6811.179 | 2811.435 | | |
| | | | | 77 | 7267.430 | 2953.019 | | |
| | | | | 78 | 7625.913 | 3114.828 | | |
| | | | | 79 | 7968.101 | 4166.588 | | |
| | | | | 80 | 8408.058 | 3438.446 | | |
| | | | | 81 | 8750.246 | 3580.029 | | |
| | | | | 82 | 9190.203 | 3640.708 | | |
| | | | | 83 | 9532.391 | 3842.969 | | |
| | | | | 84 | 9907.169 | 4004.779 | | |
| | | | | | | | 10200.473 | 4045.231 |
| 2 | 8 | 9874.580 | 4449.754 | | | | | |
| | | | | 85 | 8913.193 | 4166.588 | | |
| | | | | 86 | 7951.807 | 3802.517 | | |
| | | | | 87 | 6925.242 | 3397.994 | | |
| | | | | 88 | 5914.971 | 3054.149 | | |
| | | | | 89 | 4855.816 | 2629.400 | | |
| | | | | 90 | 3861.840 | 2326.008 | | |
| | | | | 91 | 2753.802 | 1941.711 | | |
| | | | | 92 | 1743.531 | 1577.640 | | |
| | | | | | | | 1450.227 | 1476.509 |
| 3 | 8 | 1336.164 | 1820.354 | | | | | |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | | 93 | 2525.676 | 2143.972 | | |
| | | | | 94 | 3666.304 | 2629.400 | | |
| | | | | 95 | 4741.753 | 3013.697 | | |
| | | | | 96 | 5817.203 | 3397.994 | | |
| | | | | 97 | 6957.831 | 3822.743 | | |
| | | | | 98 | 8049.575 | 4207.040 | | |
| | | | | 99 | 9108.730 | 4611.563 | | |
| | | | | 100 | 9695.338 | 4793.599 | | |
| | | | | | | | 10347.126 | 5218.348 |
| | 4 | 13 | 10721.903 | 5400.383 | | | | |
| | | | | 101 | 10086.410 | 5157.670 | | |
| | | | | 102 | 9532.391 | 4975.634 | | |
| | | | | 103 | 8848.014 | 4753.146 | | |
| | | | | 104 | 8131.048 | 4571.111 | | |
| | | | | 105 | 7495.556 | 4247.492 | | |
| | | | | 106 | 6811.179 | 4045.231 | | |
| | | | | 107 | 6273.454 | 3802.517 | | |
| | | | | 108 | 5637.961 | 3580.029 | | |
| | | | | 109 | 4969.879 | 3357.542 | | |
| | | | | 110 | 4334.386 | 3033.923 | | |
| | | | | 111 | 3666.304 | 2811.435 | | |
| | | | | 112 | 3030.812 | 2629.400 | | |
| | | | | 113 | 2346.435 | 2427.139 | | |
| | | | | | | | 1727.236 | 2204.651 |
| | 5 | 13 | 1303.575 | 2609.174 | | | | |
| | | | | 114 | 1955.362 | 2851.888 | | |
| | | | | 115 | 2590.855 | 3074.376 | | |
| | | | | 116 | 3275.232 | 3317.089 | | |
| | | | | 117 | 3894.430 | 3600.255 | | |
| | | | | 118 | 4546.217 | 3782.291 | | |
| | | | | 119 | 5198.005 | 4025.005 | | |
| | | | | 120 | 5849.792 | 4227.266 | | |
| | | | | 121 | 6468.990 | 4449.754 | | |
| | | | | 122 | 7153.367 | 4773.372 | | |
| | | | | 123 | 7772.565 | 4955.408 | | |
| | | | | 124 | 8456.942 | 5198.122 | | |
| | | | | 125 | 9076.140 | 5420.609 | | |
| | | | | 126 | 9695.338 | 5643.097 | | |
| | | | | | | | 10200.473 | 5825.133 |
| | 6 | 10 | 10200.473 | 6270.108 | | | | |
| | | | | 124 | 9516.097 | 6027.394 | | |
| | | | | 128 | 8603.594 | 5703.775 | | |
| | | | | 129 | 7642.208 | 5319.479 | | |
| | | | | 130 | 6762.295 | 5177.896 | | |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | | 131 | 5784.613 | 4773.372 | | |
| | | | | 132 | 4904.700 | 4308.171 | | |
| | | | | 133 | 4106.261 | 3964.326 | | |
| | | | | 134 | 3242.642 | 3620.482 | | |
| | | | | 135 | 2362.729 | 3317.089 | | |
| | | | | 136 | 1825.005 | 3155.280 | | |
| | | | | | | | 1075.449 | 2892.340 |
| 7 | 12 | 1385.048 | 3559.803 | | | | | |
| | | | | 137 | 2281.256 | 3842.969 | | |
| | | | | 138 | 2998.222 | 4045.231 | | |
| | | | | 139 | 3747.778 | 4348.623 | | |
| | | | | 140 | 4529.923 | 4611.563 | | |
| | | | | 141 | 5230.594 | 4854.277 | | |
| | | | | 142 | 5980.150 | 5218.348 | | |
| | | | | 143 | 6811.179 | 5461.062 | | |
| | | | | 144 | 7479.261 | 5804.906 | | |
| | | | | 145 | 8310.290 | 6047.620 | | |
| | | | | 146 | 9076.140 | 6351.013 | | |
| | | | | 147 | 9890.874 | 6674.631 | | |
| | | | | 148 | 10216.768 | 6795.988 | | |
| | | | | | | | 10673.019 | 6735.310 |



SHIP MINE CHARACTERISTIC LIST

| MINE TYPE (II) | A | B | C | D | E | F | G |
|---|---|---|---|---|---|---|---|
| 1 | .2 | .4 | 2 | 0 | .9 | .9 | 1.0 |
| 2 | .1 | .4 | 2 | 0 | .9 | .9 | .9 |
| 3 | .1 | .4 | 2 | 0 | .9 | .9 | 1.0 |
| 4 | .1 | .4 | 2 | 0 | .9 | .9 | .9 |
| 5 | .1 | .4 | 2 | 0 | .9 | .9 | .9 |
| 6 | .2 | .4 | 2 | 0 | 1.0 | .9 | 1.0 |
| 7 | .2 | .4 | 2 | 0 | 1.0 | .9 | 1.0 |
| 8 | .2 | .4 | 2 | 0 | .9 | .9 | .9 |
| 9 | .1 | .4 | 3 | 0 | 0 | 0 | .5 |
| 10 | .2 | .4 | 1 | 0 | .9 | .9 | .9 |

A = PROB OF DAMAGE ON HARDEST BOTTOM,GIVEN DEPTH LESS THAN 20FT
B = PROB OF BURYING IN SOFTEST BOTTOM
C = INDICATOR OF MOOR OR BOTTOM TYPE
D = NWL TYPE IF BOTTOM MINE DOESNOT BURY/BURIES
E = PROB OF CASE/ANCHOR SEPARATION
F = PROB OF FULL ANCHOR CABLE DEPLOYMENT
G = PROB MINE OPERATES,GIVEN IT HAS NOT BEEN DAMAGED (RELIABILITY)

SUBMARINE CHARACTERISTIC LIST

| VEHICLE TYPE | XSIT SPEED | M/L SPEED | NAVIGATION ERROR BEST | AVG | GREATEST | PMR | FTDM | PD | PKD/D | PKD/ND | PSO/KD | PR/SO | PR/UA | PA/D | PSA/A | D(100) |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 12 | 4 | 25 | 100 | 250 | .9 | 1.6 | .2 | .9 | .1 | 1.0 | 1.0 | 0 | .8 | .5 | 10000 |
| 2 | 15 | 5 | 15 | 75 | 200 | .9 | 1.5 | .2 | .9 | .1 | 0 | 0 | 1.0 | .9 | .4 | 10000 |

PMR = PROB MINE IS READY AT FIRING TIME
FTDM = FIRING TIME DELAY (MAXIMUM)
PKD/D = PROB SUBMARINE HAS KNOWLEDGE OF DETECTION,GIVEN DETECTION HAS OCCURRED
PKD/ND = PROB SUBMARINE THINKS DETECTION HAS OCCURRED,GIVEN NO DETECTION
PSO/KD = PROB SUBMARINE SUSPENDS OPS,GIVEN IT HAS KNOWLEDGE OF DETECTION
PR/SO = PROB SUBMARINE RETURNS,GIVEN IT HAD SUSPENDED OPS
PR/UA = PROB SUBMARINE RETURNS,GIVEN UNSUCCESFUL ATTACK
PA/D = PROB OF ATTACK,GIVEN DETECTION
PSA/A = PROB OF SUCCESSFUL ATTACK,GIVEN ATTACK
D(100) = DISTANCE FROM MINEFIELD PERIMETER TO 100 FATHOM CURVE

DATE-TIME FOR COMPLETION OF SUBMARINE OPERATIONS IS 200730.5

SUBMARINE LEG DESCRIPTION LIST

| LEG NO. | NO. OF MINES THIS LEG | SUB INITIAL POSIT XCOORD | SUB INITIAL POSIT YCOORD | MINE NO. | SUB FIRING POSIT XCOORD | SUB FIRING POSIT YCOORD | MINE INTENDED POSITION XCOORD | MINE INTENDED POSITION YCOORD | SUB TERMINAL POINT YCOORD | SUB TERMINAL POINT YCOORD |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 6 | 7137.072 | 6027.394 | | | | | | | |
| | | | | 149 | 7919.217 | 4126.136 | 8261.406 | 1152.891 | | |
| | | | | 150 | 8228.817 | 3296.863 | 8652.478 | 242.714 | | |
| | | | | 151 | 8473.237 | 2629.400 | 9271.676 | 566.332 | | |
| | | | | 152 | 9825.696 | 3600.255 | 9890.874 | 788.820 | | |
| | | | | 153 | 8000.691 | 4025.005 | 10558.957 | 1334.926 | | |
| | | | | 154 | 9548.686 | 4389.076 | 10705.609 | 2386.686 | | |
| | | | | | | | | | 8701.362 | 6411.691 |
| 2 | 5 | 10526.367 | 2953.019 | | | | | | | |
| | | | | 155 | 10119.000 | 2770.983 | 10119.000 | 2770.983 | | |
| | | | | 156 | 9695.338 | 2568.722 | 9695.338 | 2568.722 | | |
| | | | | 157 | 9320.560 | 2346.234 | 9320.560 | 2346.234 | | |
| | | | | 158 | 8896.899 | 2103.520 | 8896.899 | 2103.520 | | |
| | | | | 159 | 8456.942 | 1961.937 | 8456.942 | 1961.937 | | |
| | | | | | | | | | 8098.459 | 1719.223 |
| 3 | 5 | 7577.029 | 7787.070 | | | | | | | |
| | | | | 160 | 7902.923 | 6998.249 | 7642.208 | 2932.792 | | |
| | | | | 161 | 8098.459 | 6634.179 | 7805.154 | 2568.722 | | |
| | | | | 162 | 8114.754 | 6209.429 | 8098.459 | 2083.294 | | |
| | | | | 163 | 8228.817 | 5885.811 | 8603.594 | 2609.174 | | |
| | | | | 164 | 8277.700 | 5521.740 | 8440.647 | 2892.340 | | |
| | | | | | | | | | 8717.657 | 4449.754 |
| 4 | 3 | 9043.551 | 3357.542 | | | | | | | |
| | | | | 165 | 9532.391 | 3296.863 | 9532.391 | 3296.863 | | |
| | | | | 166 | 9858.285 | 3256.411 | 9858.285 | 3256.411 | | |
| | | | | 167 | 10363.420 | 3215.959 | 10363.420 | 3215.959 | | |
| | | | | | | | | | 10835.966 | 3175.506 |

SUBMARINE MINE CHARACTERISTIC LIST

| MINE TYPE (II) | A | B | C | D | E | F | G | H | I | J | K | L |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | .0 | .3 | 3 | 0 | .9 | .9 | .9 | .9 | .6 | .3 | 400 | .9 |
| 2 | .0 | .3 | 3 | 0 | .9 | .9 | .9 | .9 | .6 | .3 | 450 | .9 |
| 3 | .1 | .3 | 3 | 0 | .9 | .9 | .9 | .9 | .6 | .3 | 420 | .9 |
| 4 | .1 | .3 | 3 | 0 | .9 | .9 | .9 | .9 | .6 | .3 | 380 | .9 |
| 5 | .0 | .3 | 3 | 0 | .9 | .9 | .9 | .9 | .6 | .3 | 0 | .9 |
| 6 | .1 | .3 | 3 | 0 | .9 | .9 | .9 | .9 | .6 | .3 | 0 | .9 |
| 7 | .1 | .3 | 3 | 0 | .9 | .9 | .9 | .9 | .6 | .3 | 0 | .9 |
| 8 | .1 | .3 | 2 | 0 | .9 | .9 | .9 | .9 | .6 | .3 | 0 | .9 |

A = PROB OF DAMAGE ON HARDEST BOTTOM
B = PROB OF BURYING IN SOFTEST BOTTOM
C = INDICATOR OF MOOR OR BOTTOM TYPE
D = NWL TYPE IF BOTTOM MINE DOES NOT BURY/FUNCTIONS
E = PROB OF CASE/ANCHOR SEPARATION
F = PROB OF FULL ANCHOR CABLE DEPLOYMENT
G = PROB OF NORMAL RUN FOR A SELF-PROPELLED MINE
H = PROB OF MOTOR FAILURE, GIVEN FAILURE
I = PROB OF GYRO FAILURE, GIVEN FAILURE
J = PROB OF DEPTH FAILURE, GIVEN FAILURE
K = CEP FOR NORMAL RUN OF SELF-PROPELLED MINE
L = PROB MINE OPERATES, GIVEN IT HAS NOT BEEN DAMAGED (RELIABILITY)

MINE DESCRIPTION LIST

| MINE CASE NO. | MINE TYPE II | MINE TYPE I | MINE CASE DEPTH | ARMING DELAY | STERILE TIME | SHIP COUNTS | TYPE OF MOOR |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 0 | 0 | 688 | 93 | 3 | 0 |
| 2 | 2 | 0 | 0 | 744 | 74 | 9 | 0 |
| 3 | 3 | 0 | 0 | 476 | 90 | 3 | 0 |
| 4 | 4 | 0 | 0 | 706 | 53 | 3 | 0 |
| 5 | 5 | 0 | 0 | 952 | 57 | 2 | 0 |
| 6 | 6 | 0 | 0 | 427 | 58 | 7 | 0 |
| 7 | 7 | 0 | 0 | 892 | 56 | 8 | 0 |
| 8 | 8 | 0 | 0 | 560 | 57 | 4 | 0 |
| 9 | 9 | 0 | 0 | 572 | 91 | 3 | 0 |
| 10 | 10 | 0 | 0 | 941 | 91 | 9 | 0 |
| 11 | 11 | 0 | 0 | 170 | 50 | 7 | 0 |
| 12 | 12 | 0 | 0 | 187 | 65 | 10 | 0 |
| 13 | 7 | 0 | 0 | 919 | 50 | 11 | 0 |
| 14 | 6 | 0 | 0 | 164 | 169 | 9 | 0 |
| 15 | 13 | 0 | 0 | 152 | 115 | 8 | 0 |
| 16 | 1 | 0 | 210 | 655 | 51 | 1 | 1 |
| 17 | 14 | 0 | 0 | 552 | 98 | 9 | 0 |
| 18 | 24 | 0 | 0 | 610 | 108 | 9 | 0 |
| 19 | 3 | 0 | 0 | 105 | 93 | 17 | 0 |
| 20 | 10 | 0 | 0 | 819 | 76 | 15 | 0 |
| 21 | 12 | 0 | 0 | 896 | 99 | 6 | 0 |
| 22 | 7 | 0 | 0 | 194 | 79 | 6 | 0 |
| 23 | 16 | 0 | 0 | 664 | 85 | 5 | 0 |
| 24 | 17 | 0 | 0 | 531 | 180 | 2 | 0 |
| 25 | 5 | 0 | 0 | 32 | 95 | 8 | 0 |
| 26 | 18 | 0 | 0 | 60 | 92 | 7 | 0 |
| 27 | 12 | 0 | 0 | 18 | 159 | 27 | 0 |
| 28 | 13 | 0 | 0 | 97 | 59 | 3 | 0 |
| 29 | 13 | 0 | 0 | 64 | 88 | 8 | 0 |
| 30 | 13 | 0 | 174 | 89 | 76 | 1 | 1 |
| 31 | 19 | 0 | 0 | 941 | 137 | 23 | 0 |
| 32 | 20 | 0 | 0 | 100 | 91 | 7 | 0 |
| 33 | 21 | 0 | 0 | 557 | 85 | 3 | 0 |
| 34 | 11 | 0 | 0 | 77 | 82 | 14 | 0 |
| 35 | 22 | 0 | 0 | 656 | 122 | 4 | 0 |
| 36 | 23 | 0 | 0 | 121 | 199 | 8 | 0 |
| 37 | 24 | 0 | 0 | 153 | 57 | 6 | 0 |
| 38 | 25 | 0 | 0 | 170 | 124 | 6 | 0 |
| 39 | 10 | 0 | 0 | 283 | 73 | 2 | 0 |
| 40 | 12 | 0 | 0 | 699 | 124 | 2 | 0 |
| 41 | 15 | 0 | 0 | 341 | 98 | 8 | 0 |
| 42 | 22 | 0 | 0 | 908 | 175 | 8 | 0 |
| 43 | 23 | 0 | 0 | 149 | 62 | 9 | 0 |
| 44 | 24 | 0 | 0 | 246 | 94 | 7 | 0 |
| 45 | 25 | 0 | 0 | 366 | 99 | 10 | 0 |
| 46 | 26 | 0 | 0 | 415 | 138 | 2 | 0 |
| 47 | 12 | 0 | 0 | 7 | 97 | 18 | 0 |
| 48 | 26 | 0 | 0 | 574 | 55 | 5 | 0 |
| 49 | 14 | 0 | 0 | 512 | 67 | 9 | 0 |
| 50 | 15 | 0 | 0 | 373 | 158 | 9 | 0 |
| 51 | 23 | 0 | 0 | 884 | 52 | 3 | 0 |
| 52 | 8 | 0 | 0 | 114 | 65 | 7 | 0 |
| 53 | 2 | 0 | 0 | 893 | 196 | 8 | 0 |
| 54 | 3 | 0 | 0 | 831 | 75 | 9 | 0 |
| 55 | 10 | 0 | 0 | 961 | 94 | 16 | 0 |
| 56 | 1 | 0 | 0 | 232 | 65 | 6 | 0 |
| 57 | 17 | 0 | 0 | 885 | 57 | 9 | 0 |
| 58 | 19 | 0 | 0 | 834 | 97 | 3 | 0 |
| 59 | 9 | 0 | 0 | 13 | 125 | 20 | 0 |
| 60 | 6 | 0 | 0 | 301 | 140 | 7 | 0 |
| 61 | 7 | 0 | 0 | 903 | 78 | 9 | 0 |
| 62 | 12 | 0 | 0 | 265 | 88 | 5 | 0 |
| 63 | 10 | 0 | 0 | 24 | 170 | 4 | 0 |
| 64 | 5 | 0 | 140 | 135 | 20 | 1 | 1 |
| 65 | 7 | 0 | 82 | 180 | 19 | 1 | 1 |
| 66 | 10 | 0 | 0 | 193 | 999 | 1 | 1 |
| 67 | 3 | 0 | 70 | 893 | 88 | 1 | 2 |

RUN NO. 26  IR FOR THIS RUN IS  54880769

DETAILED NARRATIVE

| MINE NO. | TIME LAYED | TIME ACTIVE | XCOORD (YDS) | YCOORD (YDS) | CASE DEPTH (FT) | REMARKS |
|---|---|---|---|---|---|---|
| 149 | 160842.5 | 161142.5 | 9181.873 | 842.700 | 210.19 | |
| 151 | 160851.6 | 161151.6 | 8536.183 | 516.263 | 192.11 | |
| 152 | 160901.5 | 161834.5 | 9629.014 | 6141.716 | 231.78 | MOTOR FAILURE. |
| 153 | 160912.6 | 161153.6 | 10784.961 | 1596.934 | 148.86 | |
| 154 | 160922.0 | 161545.0 | 10537.421 | 2340.911 | 173.74 | BURIED. |
| 155 | 161000.8 | 161729.8 | 10121.841 | 2861.498 | 192.95 | |
| 156 | 161004.0 | 162007.0 | 9603.965 | 2549.883 | 202.41 | |
| 157 | 161006.2 | 161917.2 | 9328.564 | 2446.330 | 204.42 | |
| 158 | 161009.1 | 170017.1 | 8950.209 | 2138.655 | 203.17 | |
| 159 | 161011.8 | 161711.8 | 8594.963 | 1992.331 | 29.17 | |
| 160 | 180942.3 | 182222.3 | 8190.420 | 7053.138 | 259.47 | GYRO FAILURE. |
| 161 | 180945.4 | 0 | 8222.215 | 2603.207 | 222.83 | UNRELIABLE. |
| 162 | 180948.5 | 190149.5 | 7817.592 | 1898.838 | 225.63 | BURIED. |
| 163 | 180951.1 | 0 | 8008.523 | 8081.707 | 280.41 | UNRELIABLE. DEPTH FAILURE. |
| 164 | 180953.8 | 190225.8 | 8255.621 | 3660.337 | 239.70 | |
| 165 | 200305.7 | 200605.7 | 9402.527 | 3328.325 | 221.27 | CASE/ANCHOR SEPARATION FAILURE. |
| 166 | 200308.2 | 201440.2 | 9889.044 | 3139.396 | 214.64 | |
| 167 | 200311.9 | 200611.9 | 10432.000 | 3161.787 | 6.45 | |

| | | | | | | |
|---|---|---|---|---|---|---|
| 64 | 200731.5 | 200946.5 | 2114.989 | 1191.531 | 2.42 | |
| 65 | 200732.6 | | 2496.748 | 1429.807 | 129.61 | CASE/ANCHOR SEPARATION FAILURE. DID NOT ARM DUE TO CASE/ANCHOR SEPARATION FAILURE. |
| 66 | 200733.6 | 201746.6 | 2875.653 | 1367.859 | 2.00 | |
| 67 | 200734.7 | 202227.7 | 3246.959 | 1522.943 | 23.79 | |
| 68 | 200735.7 | 202358.7 | 3719.484 | 1723.527 | 6.79 | |
| 69 | 200736.6 | 0 | 3992.162 | 1871.736 | 2.00 | UNRELIABLE. |
| 70 | 200737.6 | 200906.6 | 4490.007 | 2012.575 | 2.00 | |
| 71 | 200738.7 | 201466.7 | 4782.413 | 2132.115 | 3.36 | |
| 72 | 200739.6 | 201039.6 | 5124.311 | 2261.826 | 63.00 | |
| 73 | 200740.6 | 201550.6 | 5633.306 | 2483.894 | 206.29 | |
| 74 | 200741.9 | 201643.9 | 5993.114 | 2580.777 | 6.03 | |
| 75 | 200743.0 | 0 | 6484.532 | 2620.931 | 117.97 | UNRELIABLE. |
| 76 | 200743.9 | 202028.9 | 6875.743 | 2758.968 | 22.29 | |
| 77 | 200745.0 | 201038.0 | 7266.030 | 2917.910 | 239.42 | |
| 78 | 200746.0 | 201046.0 | 7557.700 | 3124.600 | 26.00 | |
| 79 | 200748.7 | 201219.7 | 7799.604 | 4243.575 | 15.12 | |
| 80 | 200750.8 | 201409.8 | 8452.826 | 3326.587 | 13.63 | |
| 81 | 200751.8 | 210027.8 | 8721.343 | 3647.703 | 9.35 | |
| 82 | 200752.9 | 201350.9 | 9140.330 | 3744.086 | 6.07 | |
| 83 | 200753.8 | 0 | 9581.667 | 3817.024 | -3.11 | UNRELIABLE. MINE IS WATCHING. |
| 84 | 200754.8 | 201054.8 | 9910.805 | 4114.849 | 53.00 | |
| 85 | 200759.5 | 201405.3 | 9155.847 | 4077.197 | 233.13 | |
| 86 | 200801.9 | 201539.9 | 7895.026 | 3724.821 | 8.67 | |
| 87 | 200804.6 | 202104.6 | 6772.989 | 3334.730 | 12.00 | |
| 88 | 200807.2 | 201641.2 | 5945.819 | 3210.943 | 21.00 | |
| 89 | 200810.1 | 201128.1 | 4712.299 | 2528.054 | -5.20 | MINE IS WATCHING. |
| 90 | 200812.4 | 202121.4 | 3924.121 | 2333.678 | 63.63 | |
| 91 | 200815.4 | 201124.5 | 2900.956 | 1935.698 | 158.74 | |

| | | | | | | |
|---|---|---|---|---|---|---|
| 92 | [illegible] | 201118.2 | 1938.071 | 1569.185 | 31.00 | |
| 114 | 200902.6 | 201202.6 | 1946.301 | 2013.729 | 9.75 | |
| 115 | 200905.1 | 201202.1 | 2606.530 | 3035.436 | 9.11 | |
| 116 | 200907.8 | 201233.8 | 3239.853 | 3273.013 | 11.72 | |
| 117 | 200910.3 | 201210.3 | 3945.181 | 3645.174 | 30.00 | |
| 118 | 200912.8 | [illegible] | 4572.645 | 3772.342 | 14.89 | UNRELIABLE. |
| 119 | 200915.4 | 201225.4 | 5155.193 | 4030.930 | 273.33 | |
| 120 | 200917.9 | 201940.9 | 5802.364 | 4259.555 | 23.59 | |
| 121 | 200920.3 | 200924.3 | 6411.336 | 4417.174 | 8.35 | |
| 122 | 200923.1 | 201326.1 | 7112.071 | 4779.683 | 258.25 | CASE/ANCHOR SEPARATION FAILURE. |
| 123 | 200925.5 | 210017.5 | 7730.326 | 4896.739 | .63 | |
| 124 | 200928.2 | 201255.2 | 8479.093 | 5221.977 | 246.55 | |
| 125 | 200930.6 | 201230.6 | 9066.419 | 5369.687 | 27.00 | |
| 126 | 200933.1 | 201344.1 | 9690.238 | 5606.988 | 6.74 | |
| 124 | 200939.4 | 201306.4 | 9607.625 | 5960.664 | 231.42 | |
| 128 | 200943.0 | 201338.0 | 8606.141 | 5557.597 | 6.13 | |
| 129 | 200946.8 | 201343.8 | 7670.262 | 5208.291 | 251.45 | |
| 130 | 200950.1 | 201250.1 | 6764.716 | 5286.796 | 51.00 | |
| 131 | 200954.0 | 201502.0 | 5874.930 | 4808.977 | 9.67 | |
| 132 | 200957.7 | 201414.7 | 4985.611 | 4254.983 | 222.88 | MOORING CABLE DID NOT DEPLOY FULLY. |
| 133 | 201000.9 | 202029.9 | 4209.052 | 3942.021 | 268.37 | |
| 134 | 201004.4 | 201442.4 | 3574.565 | 3682.795 | 3.23 | |
| 135 | 201007.2 | 201520.8 | 2829.999 | 3186.565 | .68 | |
| 136 | 201009.9 | 201150.9 | 1941.995 | 3239.885 | 10.32 | |
| 137 | 201012.1 | 201708.1 | 2554.277 | 3751.441 | 278.77 | |
| 138 | 201021.9 | 201721.9 | 2959.315 | 3938.363 | 47.00 | |
| 139 | 201024.9 | 202112.8 | 3945.284 | 4390.028 | 5.26 | |
| 140 | 201027.9 | 201914.9 | 4615.193 | 4689.795 | 284.41 | |

| | | | | | | |
|---|---|---|---|---|---|---|
| 141 | 201030.6 | 202227.6 | [illegible] | 4711.980 | 12.46 | |
| 142 | [illegible] | 201555.7 | 5956.897 | 5098.730 | 9.96 | |
| 143 | 201036.9 | [illegible] | 6[illegible]22.195 | 5541.840 | -1.61 | MINE IS WATCHING. |
| 144 | 201039.7 | [illegible] | 7478.044 | 5950.378 | 6.37 | |
| 145 | [illegible] | [illegible] | 8497.360 | 5997.983 | 254.73 | |
| 146 | 201045.9 | [illegible] | 9126.243 | 6296.151 | 35.00 | |
| 147 | 201049.2 | [illegible] | 9987.343 | 6779.087 | 236.76 | CASE/ANCHOR SEPARATION FAILURE. |
| 148 | [illegible] | 210134.5 | [illegible] | 6860.516 | 116.52 | |
| 1 | [illegible] | [illegible] | 15536.824 | 6768.967 | 23.11 | BURIED. |
| 2 | [illegible] | [illegible] | 15[illegible]0.241 | 6814.374 | 65.19 | UNRELIABLE. |
| 3 | [illegible] | [illegible] | 14[illegible]4.817 | 6958.436 | 96.97 | DAMAGED ON WATER IMPACT. BURIED. |
| 4 | [illegible] | [illegible] | 13947.602 | 6879.772 | 116.00 | |
| 5 | [illegible] | [illegible] | 12893.103 | 6952.005 | 185.82 | |
| 6 | 221551.9 | 222258.9 | 12113.634 | 6685.494 | 185.59 | BURIED. |
| 7 | 221552.6 | 230644.0 | 11611.142 | 6694.755 | 198.59 | |
| 8 | [illegible] | [illegible] | [illegible]73.213 | 6806.202 | 228.29 | CHUTE DID NOT DEPLOY. UNRELIABLE. |
| 9 | 221555.0 | 230127.0 | [illegible]37.834 | 6224.617 | 22.16 | |
| 10 | 221555.1 | 230736.1 | 14415.165 | 6745.271 | 93.22 | |
| 11 | 221555.2 | 221845.2 | 13900.311 | 6273.018 | 107.04 | |
| 12 | 221555.3 | 221902.3 | 13461.016 | 6261.597 | 124.99 | |
| 13 | 221555.4 | 230714.4 | 12962.702 | 6530.729 | 152.37 | |
| 14 | 221555.5 | [illegible] | 11931.900 | 6617.400 | 198.84 | DAMAGED ON WATER IMPACT. |
| 15 | 221555.6 | 221927.6 | 10943.128 | 6366.968 | 210.19 | |
| 16 | 221555.7 | 230250.7 | 10661.162 | 6031.830 | .09 | |
| 17 | 221558.7 | 222150.7 | 15473.400 | 5830.422 | 9.20 | |
| 18 | 221558.8 | [illegible] | 14436.423 | 6142.096 | 81.08 | |
| 19 | 221558.9 | [illegible] | [illegible]15.583 | [illegible]196.895 | 96.41 | CHUTE DID NOT DEPLOY. |

| | | | | | | |
|---|---|---|---|---|---|---|
| 20 | [illegible] | [illegible] | 13309.604 | 6152.622 | 131.43 | DAMAGED ON WATER IMPACT. |
| 21 | [illegible] | [illegible] | 12476.053 | 6163.049 | 162.60 | |
| 22 | [illegible] | [illegible] | 11753.381 | 6067.707 | 186.87 | UNRELIABLE. |
| 23 | [illegible] | [illegible] | 10937.212 | 5903.715 | 202.83 | |
| 24 | [illegible] | [illegible] | 9017.098 | 6058.565 | 243.78 | BURIED.<br>UNRELIABLE. |
| 35 | [illegible] | [illegible] | 13481.056 | 5097.687 | 99.90 | BURIED. |
| 36 | 221608.5 | 221809.5 | 12989.948 | 5359.647 | 126.33 | CHUTE DID NOT DEPLOY.<br>BURIED. |
| 41 | 221612.8 | 222153.8 | 14851.949 | 4760.103 | 43.78 | |
| 43 | 221613.0 | 221842.0 | 13165.639 | 4684.565 | 109.12 | CHUTE DID NOT DEPLOY.<br>BURIED. |
| 44 | 221613.1 | 222019.1 | 12954.295 | 4709.936 | 118.04 | BURIED. |
| 45 | 221613.2 | 222219.2 | 12122.245 | 4820.531 | 153.92 | BURIED. |
| 47 | 221613.4 | 221620.4 | 10594.176 | 4736.803 | 204.08 | |
| 49 | 221613.6 | 222125.1 | 13035.384 | 4055.517 | 102.34 | |
| 50 | 221613.1 | 222226.1 | 14337.469 | 5051.678 | 72.57 | |
| 53 | 221613.3 | 230706.3 | 11209.538 | 5148.967 | 187.76 | BURIED. |
| 54 | 221613.4 | 230604.4 | 9896.506 | 4319.897 | 220.70 | BURIED. |
| 55 | 221613.5 | 230814.5 | 9813.542 | 6111.675 | 227.76 | |
| 56 | 221618.1 | [illegible] | 15313.326 | 3741.904 | -46.35 | ON THE BEACH. |
| 59 | 221618.3 | 221631.3 | 13143.522 | 2859.811 | 78.23 | |
| 60 | 221618.4 | 222119.4 | 13953.263 | 3306.346 | 51.99 | |
| 61 | 221618.5 | 230721.5 | 11084.574 | 5716.013 | 197.32 | CHUTE DID NOT DEPLOY.<br>BURIED. |
| 62 | 221618.6 | 222041.6 | 10088.326 | 3988.960 | 212.11 | BURIED. |

MINES LAYED 123 MINES EFFECTIVE 106 MINES BURIED 15 MINES NOT BURIED 0 MINES COMPROMISED 4

SUMMARY OF RESULTS

| RUN NO. | MINES LAYED | MINES EFFECTIVE | MINES BURIED | MINES NOT BURIED | MINES COMPROMISED |
|---|---|---|---|---|---|
| 1 | 147 | 130 | 24 | 0 | 8 |
| 2 | 129 | 104 | 25 | 0 | 4 |
| 3 | 119 | 93 | 22 | 0 | 4 |
| 4 | 126 | 103 | 26 | 0 | 7 |
| 5 | 119 | 95 | 18 | 0 | 6 |
| 6 | 156 | 129 | 20 | 0 | 7 |
| 7 | 110 | 99 | 19 | 0 | 5 |
| 8 | 110 | 89 | 22 | 0 | 6 |
| 9 | 157 | 133 | 26 | 0 | 5 |
| 10 | 144 | 120 | 20 | 0 | 6 |
| 11 | 135 | 108 | 20 | 0 | 7 |
| 12 | 126 | 107 | 18 | 0 | 6 |
| 13 | 139 | 115 | 22 | 0 | 9 |
| 14 | 132 | 110 | 21 | 0 | 6 |
| 15 | 147 | 119 | 21 | 0 | 9 |
| 16 | 125 | 106 | 24 | 0 | 6 |
| 17 | 136 | 115 | 18 | 0 | 5 |
| 18 | 157 | 137 | 21 | 0 | 5 |
| 19 | 101 | 75 | 24 | 0 | 6 |
| 20 | 104 | 88 | 21 | 0 | 4 |
| 21 | 162 | 136 | 27 | 0 | 7 |
| 22 | 95 | 73 | 20 | 0 | 7 |
| 23 | 128 | 111 | 19 | 0 | 5 |
| 24 | 133 | 113 | 23 | 0 | 6 |
| 25 | 115 | 91 | 18 | 0 | 8 |
| 26 | 123 | 106 | 15 | 0 | 4 |
| 27 | 147 | 120 | 17 | 0 | 8 |
| 28 | 159 | 131 | 30 | 0 | 6 |
| 29 | 150 | 120 | 17 | 0 | 9 |
| 30 | 134 | 108 | 24 | 0 | 6 |
| 31 | 128 | 109 | 21 | 0 | 5 |
| 32 | 123 | 97 | 25 | 0 | 8 |
| 33 | 148 | 127 | 25 | 0 | 8 |
| 34 | 136 | 117 | 16 | 0 | 5 |
| 35 | 157 | 129 | 27 | 0 | 8 |
| 36 | 157 | 131 | 22 | 0 | 6 |
| 37 | 160 | 129 | 20 | 0 | 7 |
| 38 | 151 | 123 | 22 | 0 | 5 |
| 39 | 153 | 124 | 19 | 0 | 9 |
| 40 | 112 | 91 | 16 | 0 | 5 |
| 41 | 154 | 126 | 22 | 0 | 6 |
| 42 | 150 | 119 | 15 | 0 | 10 |
| 43 | 138 | 132 | 22 | 0 | 7 |
| 44 | 121 | 106 | 21 | 0 | 3 |
| 45 | 139 | 120 | 23 | 0 | 2 |
| 46 | 132 | 123 | 24 | 0 | 7 |
| 47 | 133 | 110 | 25 | 0 | 5 |
| 48 | 146 | 114 | 20 | 0 | 11 |
| 49 | 101 | 128 | 22 | 0 | 8 |
| 50 | 137 | 114 | 15 | 0 | 6 |
| 51 | 125 | 102 | 23 | 0 | 8 |
| 52 | 128 | 114 | 17 | 0 | 7 |
| 53 | 139 | 116 | 23 | 0 | 5 |
| 54 | 130 | 110 | 24 | 0 | 7 |
| 55 | 142 | 113 | 25 | 0 | 6 |
| 56 | 171 | 138 | 21 | 0 | 4 |
| 57 | 146 | 123 | 18 | 0 | 7 |
| 58 | 134 | 109 | 20 | 0 | 5 |
| 59 | 145 | 117 | 18 | 0 | 5 |
| 60 | 130 | 123 | 15 | 0 | 11 |
| 61 | 146 | 124 | 18 | 0 | 10 |
| 62 | 141 | 121 | 18 | 0 | 7 |
| 63 | 116 | 99 | 16 | 0 | 2 |
| 64 | 112 | 101 | 13 | 0 | 3 |
| 65 | 154 | 129 | 15 | 0 | 7 |
| 66 | 163 | 142 | 23 | 0 | 5 |
| 6 | 129 | 105 | 24 | 0 | 6 |

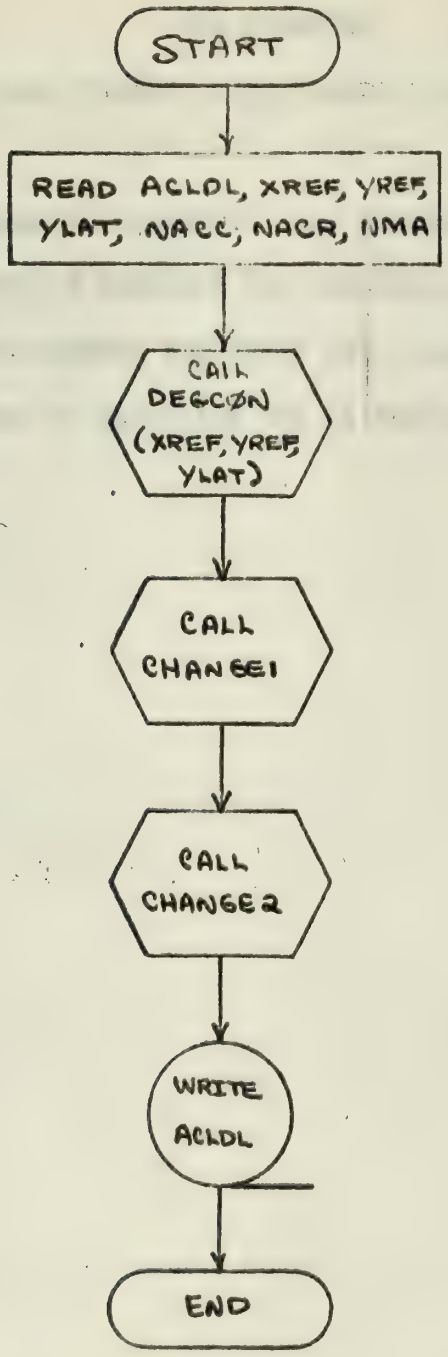
| | | |

## APPENDIX XIV

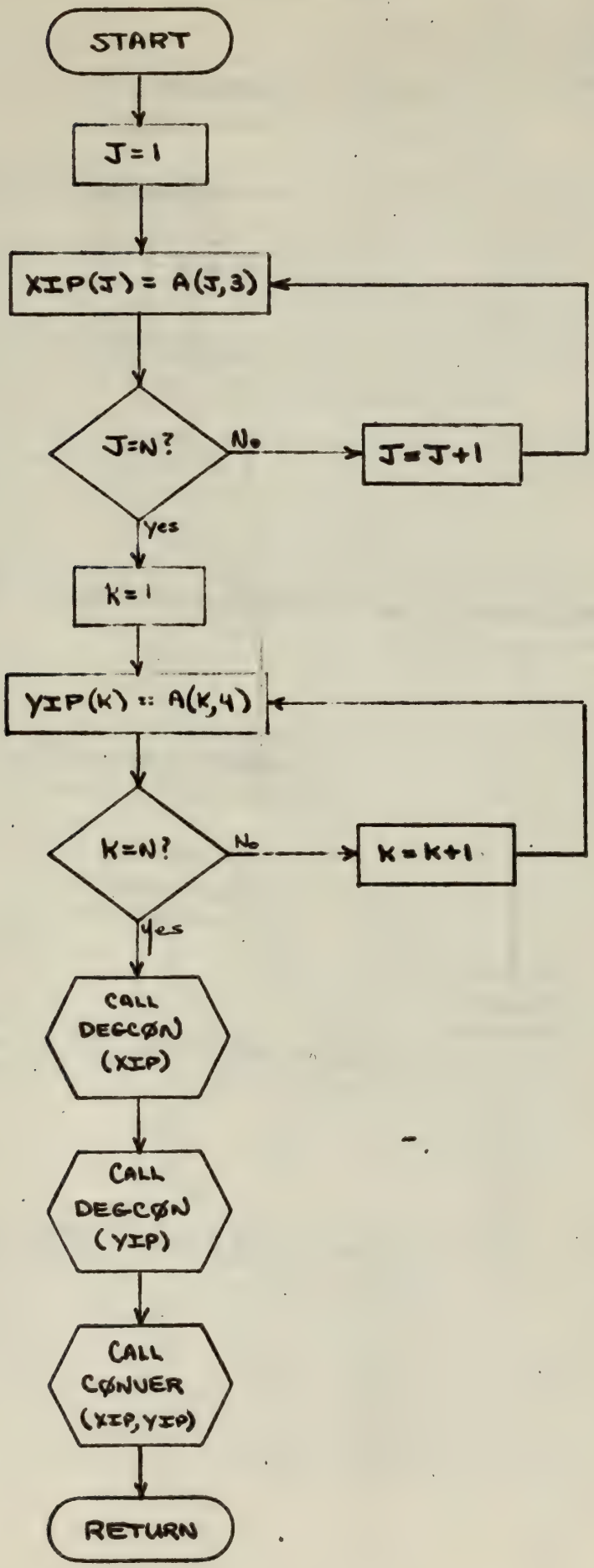
## FLOWCHARTS FOR PROGRAMS GRID, SURFIT, AND MINDEL1

This appendix contains the flowcharts of programs GRID, SURFIT, and MINDEL1. These flowcharts are included to aid the user in working with those programs. The flowchart symbols shown in Figure 18 and the approach described at the beginning of Appendix IV were used.

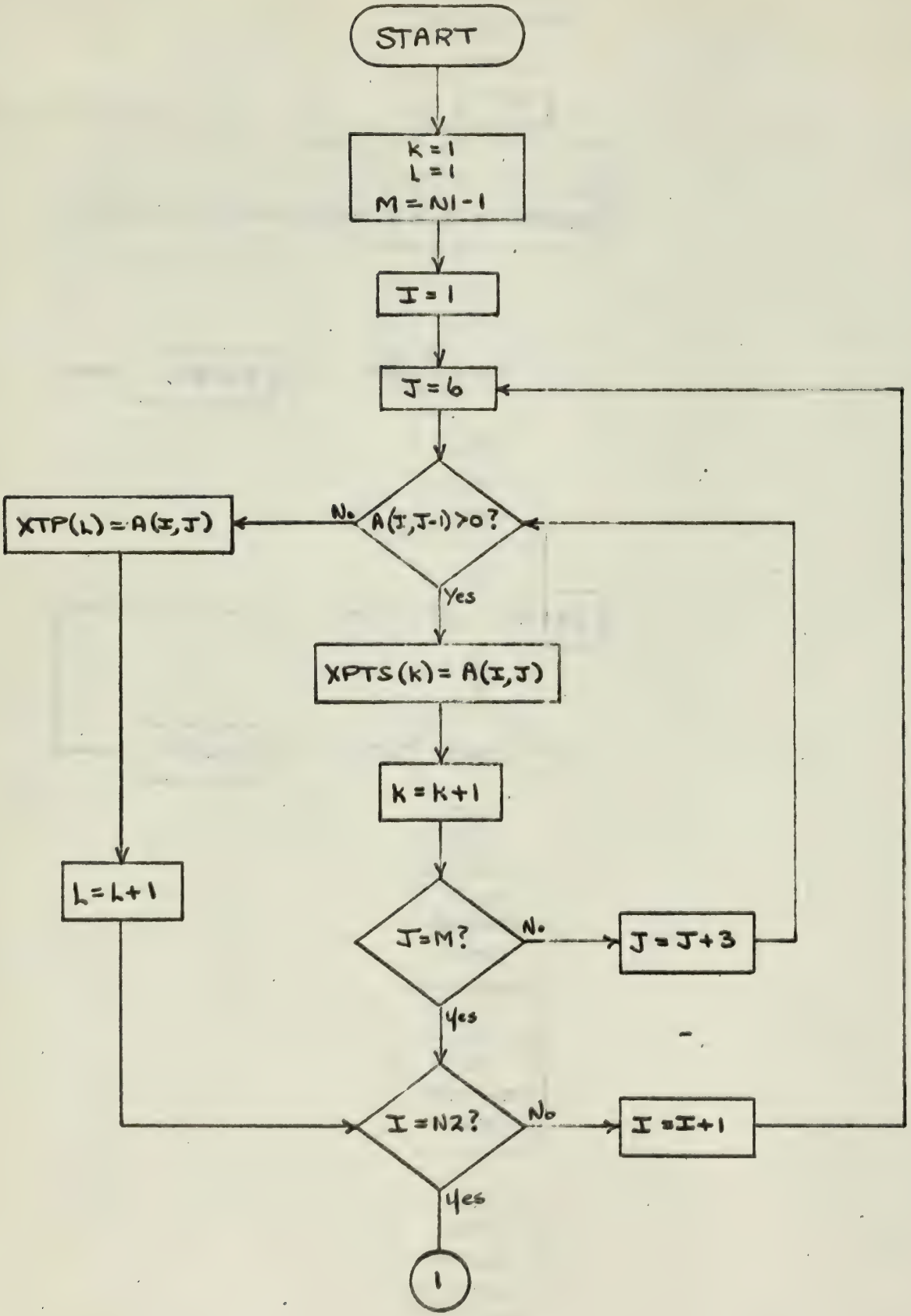
PROGRAM GRID



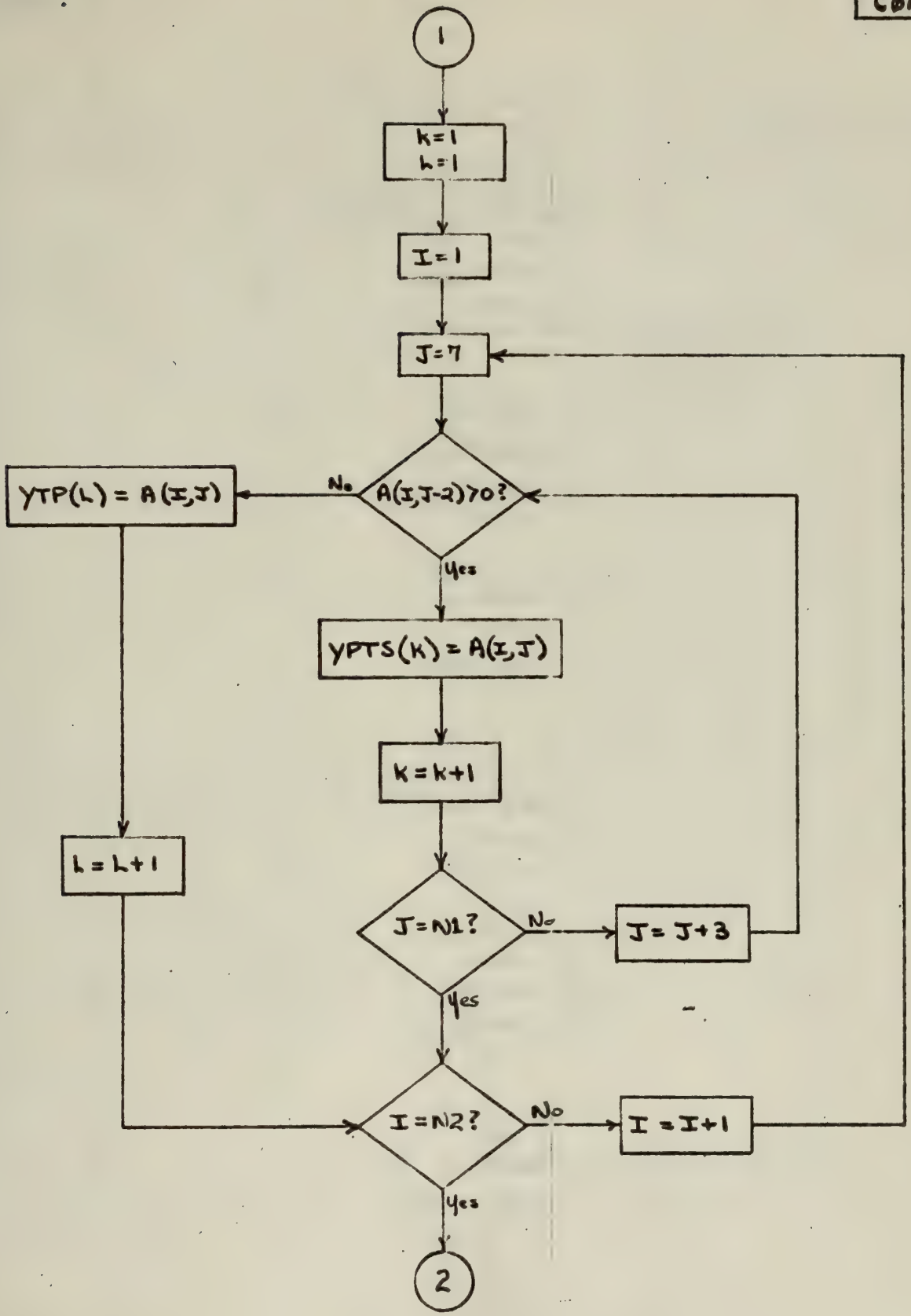
SUBRØUTINE CHANGE1 .



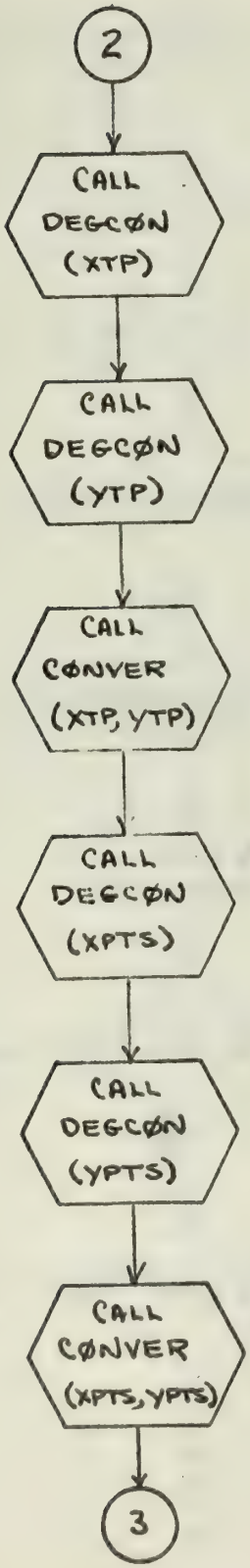
# SUBROUTINE CHANGE: 2

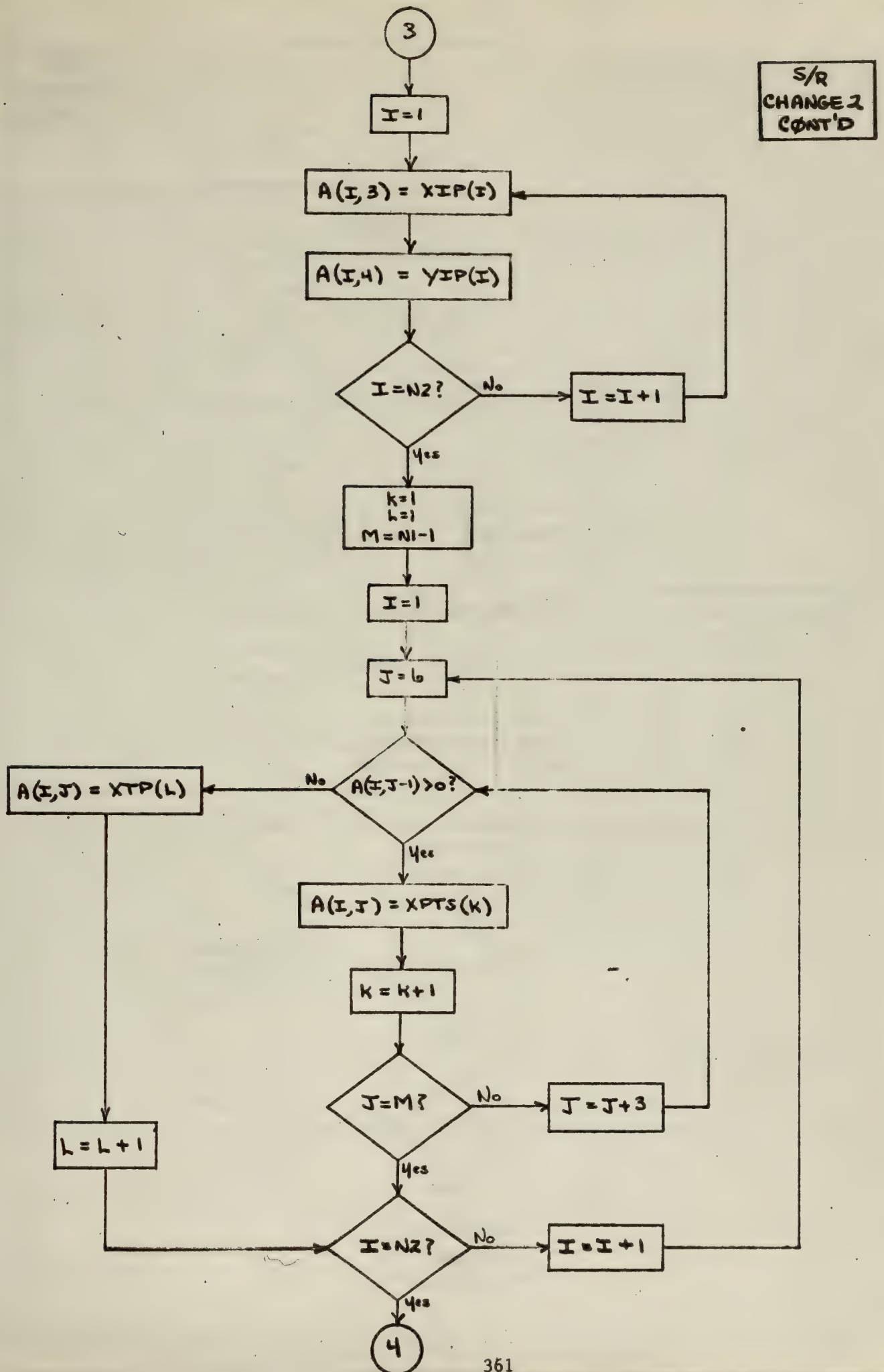


S/R CHANGE 2 CONT'D

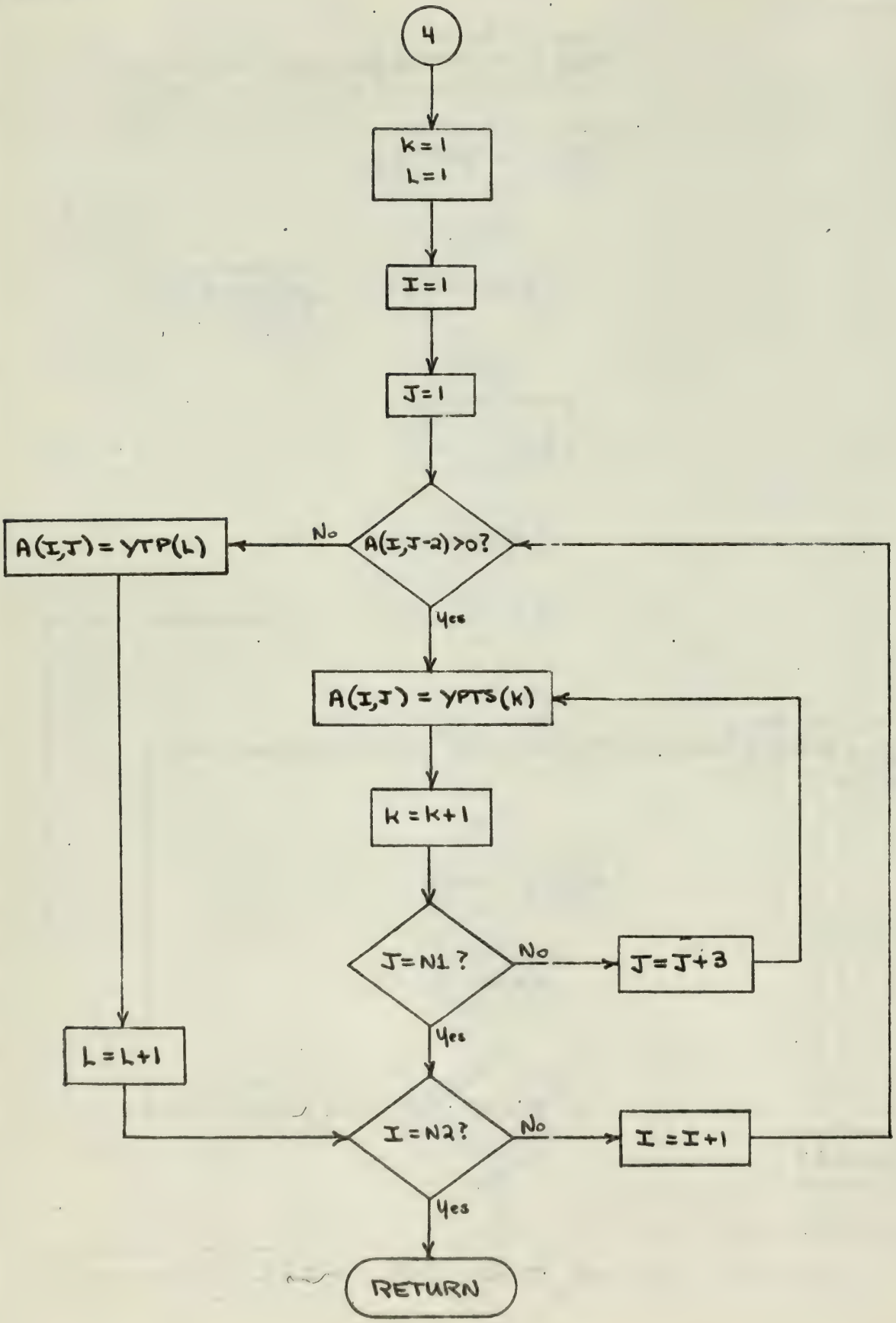


S/R CHANGE 2 CØNT'D

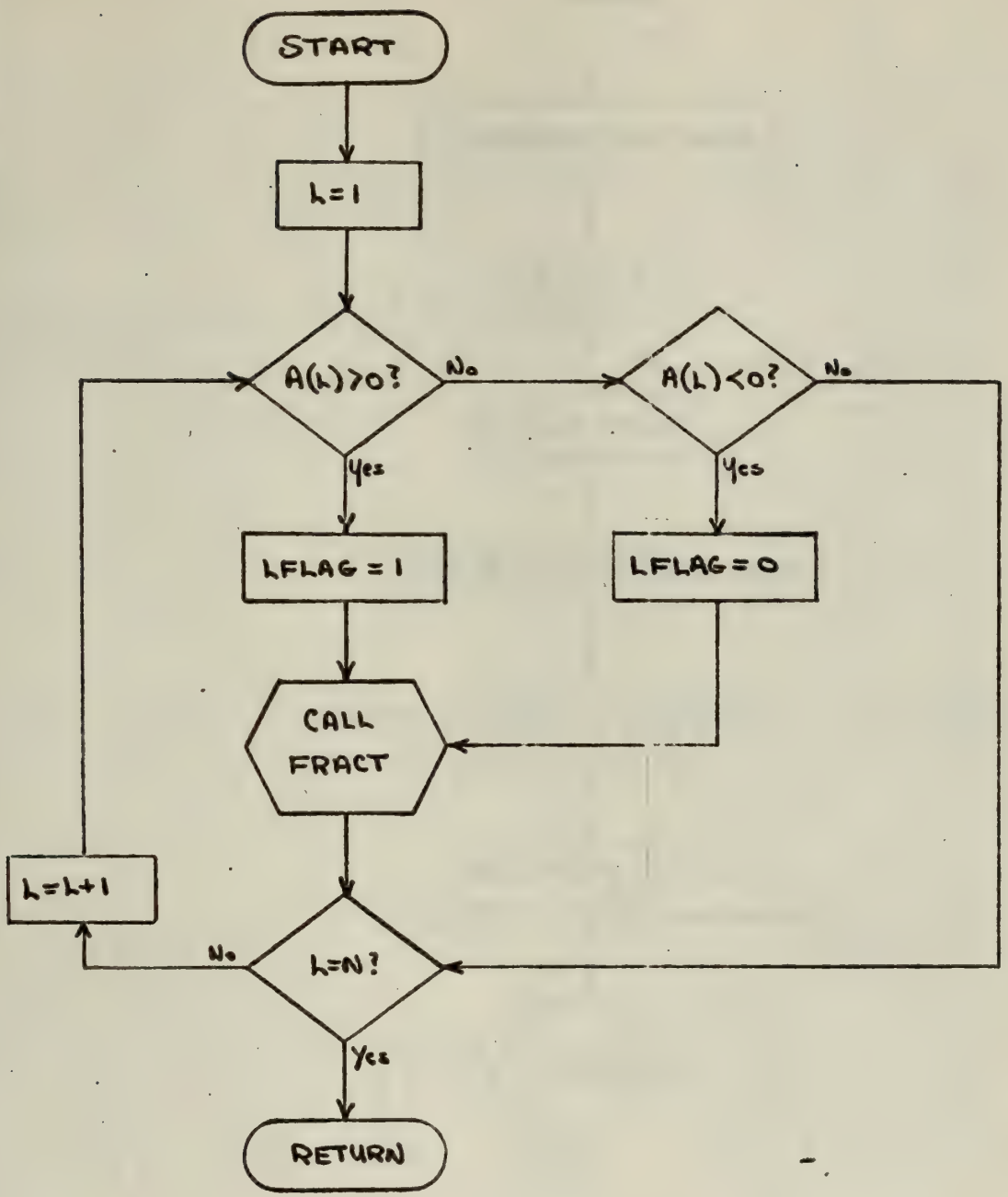



3
S/R CHANGE 2 CONT'D
I=1
A(I,3) = XIP(I)
A(I,4) = YIP(I)
I=N2?
No
I=I+1
Yes
K=1
L=1
M=N1-1
I=1
J=1
A(I,J-1)>0?
No
A(I,J) = XTP(L)
Yes
A(I,J) = XPTS(K)
K=K+1
J=M?
No
J=J+3
L=L+1
Yes
I=N2?
No
I=I+1
Yes
4

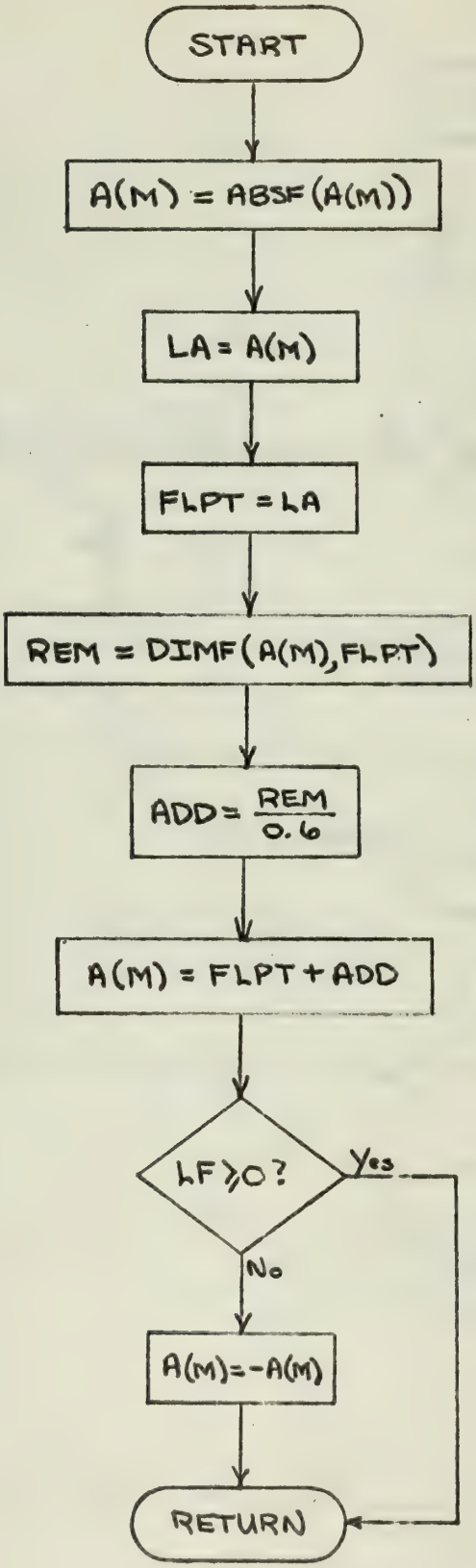
S/R
CHANGE 2
CØNT'D



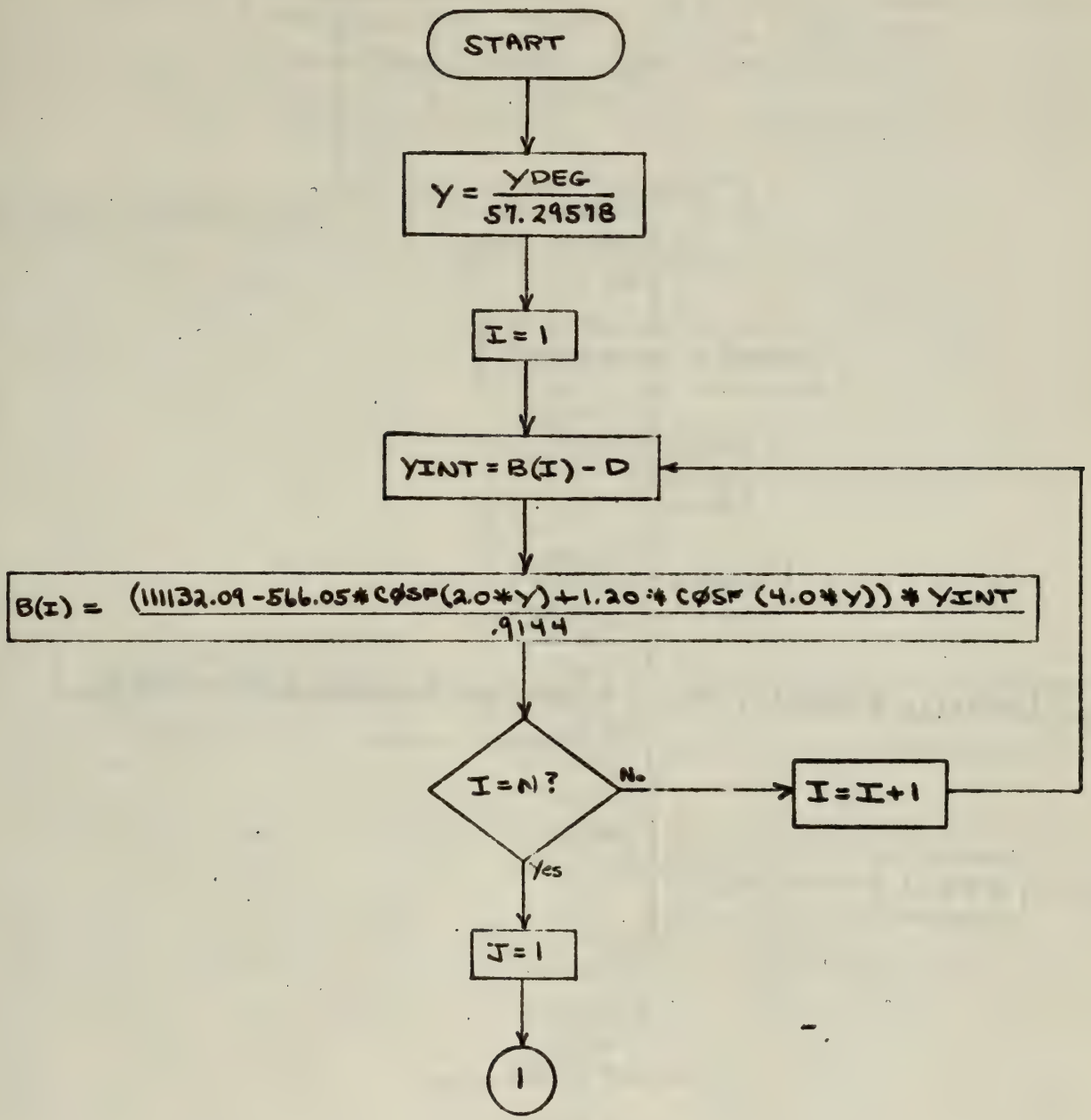
# SUBROUTINE DEGCON

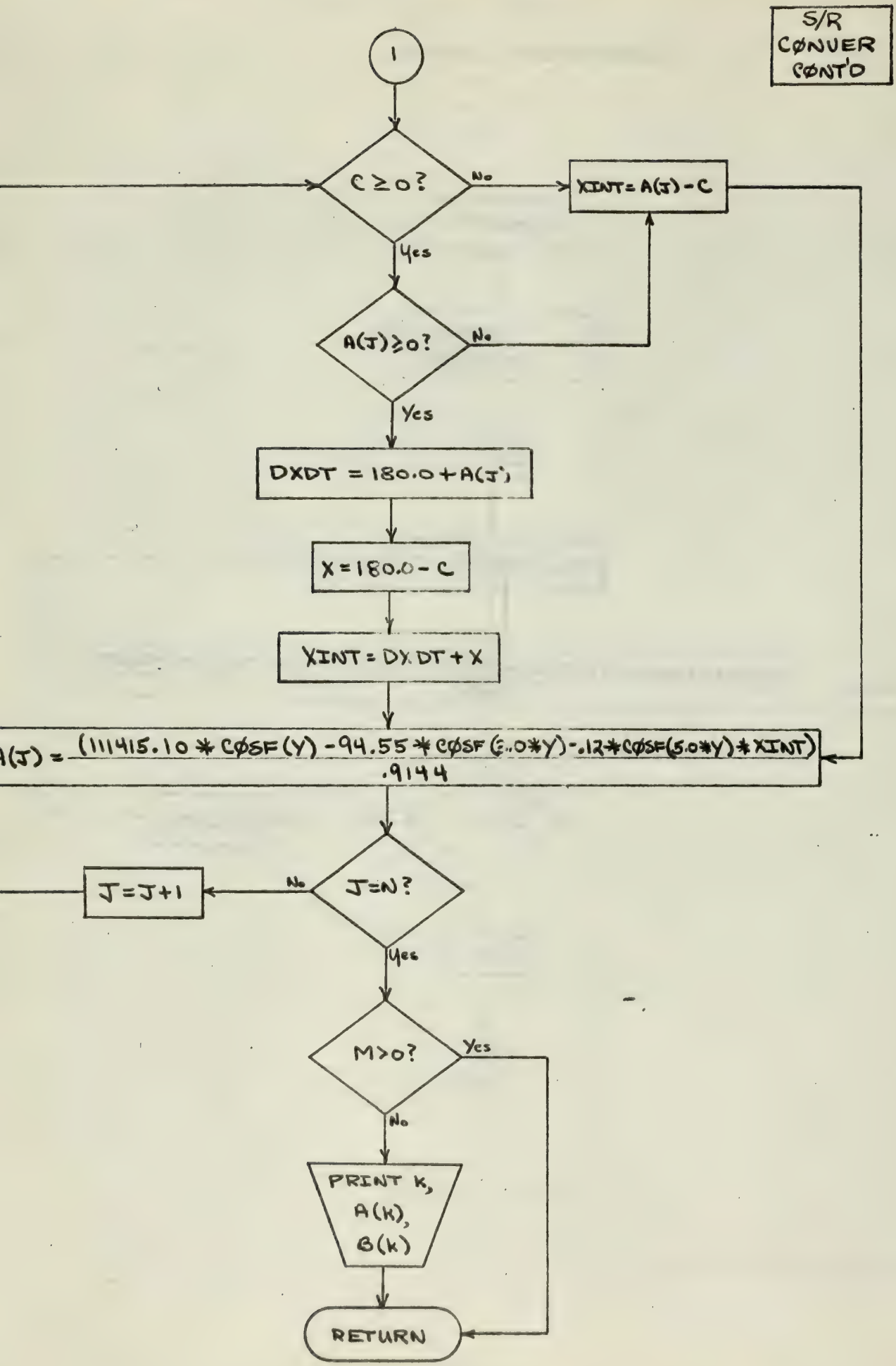


# SUBRØUTINE FRACT

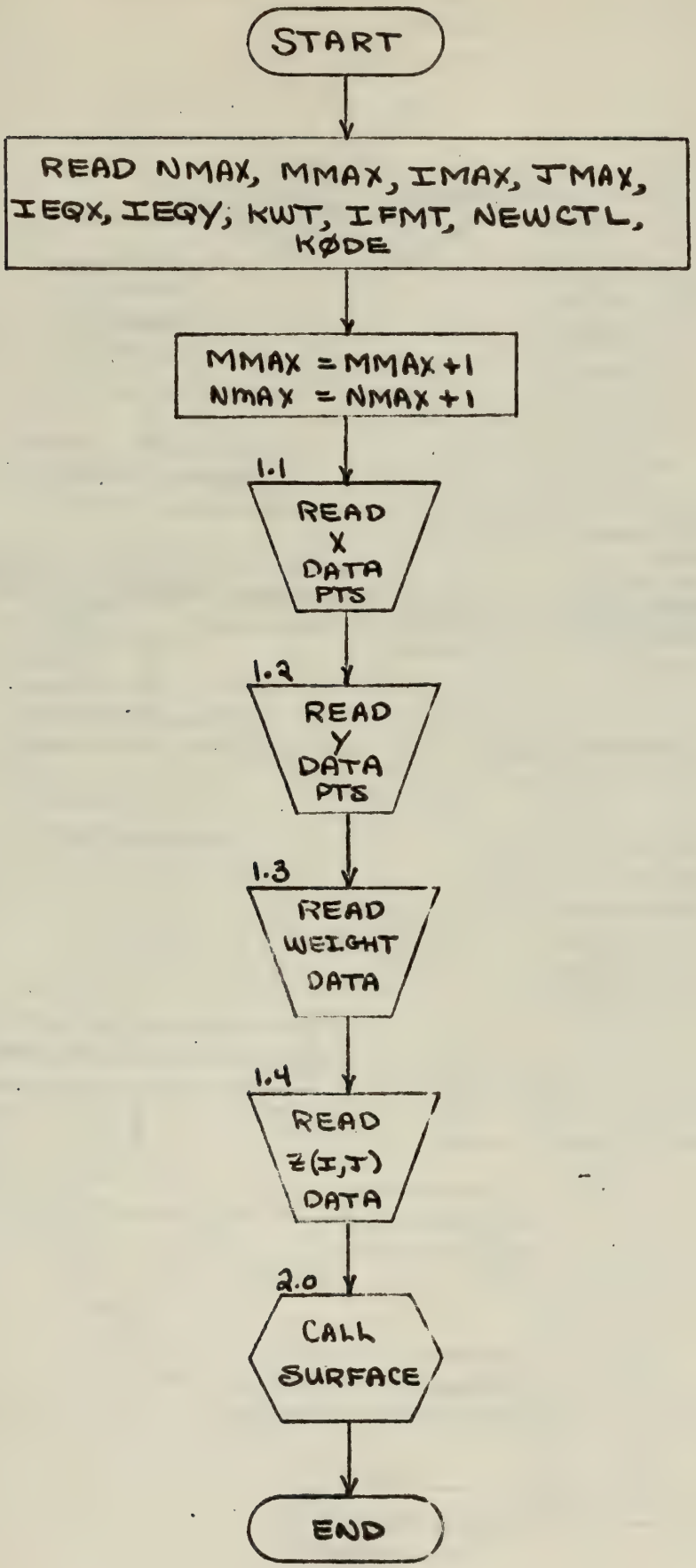


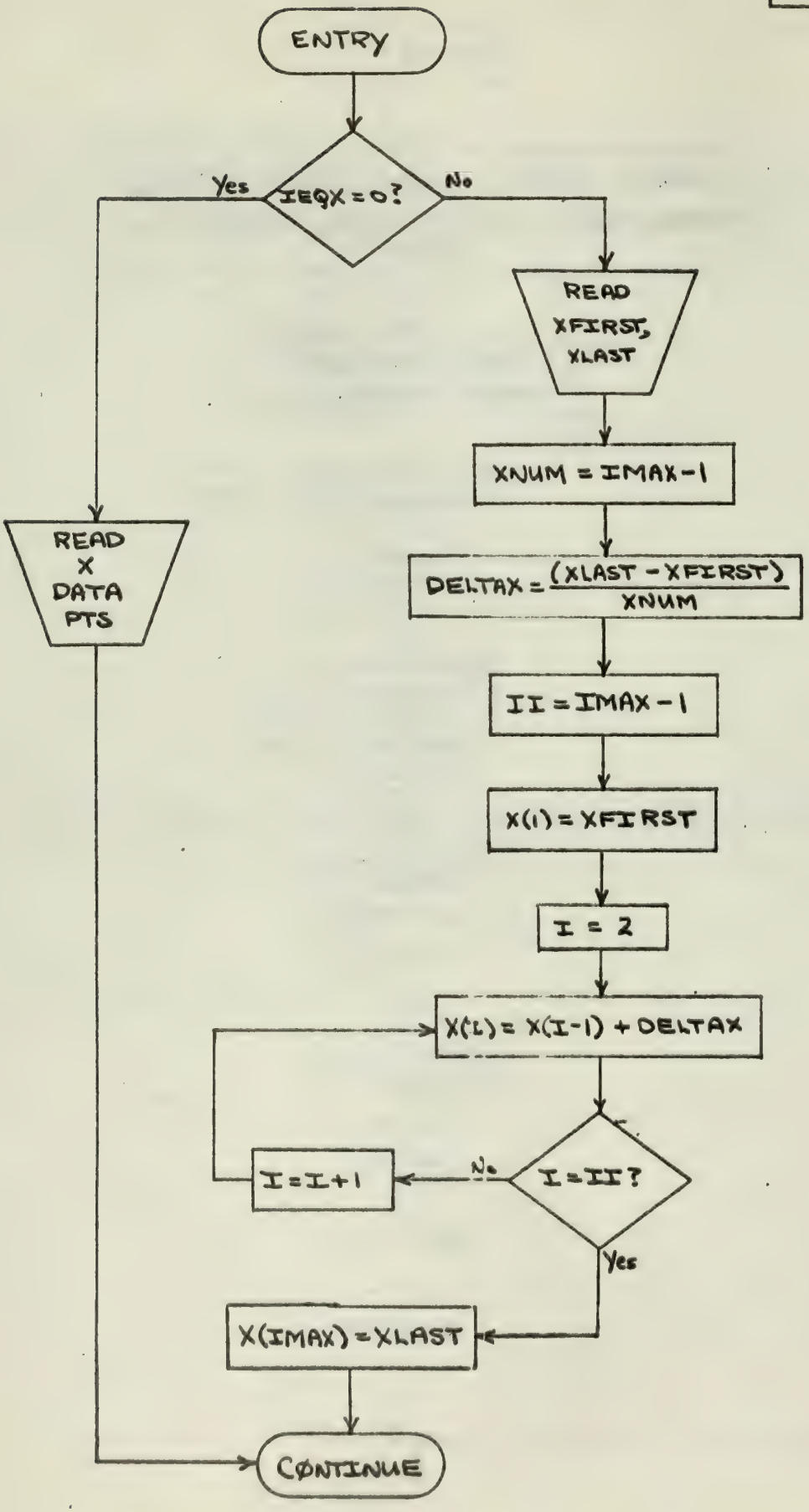
# SUBROUTINE CONVER:

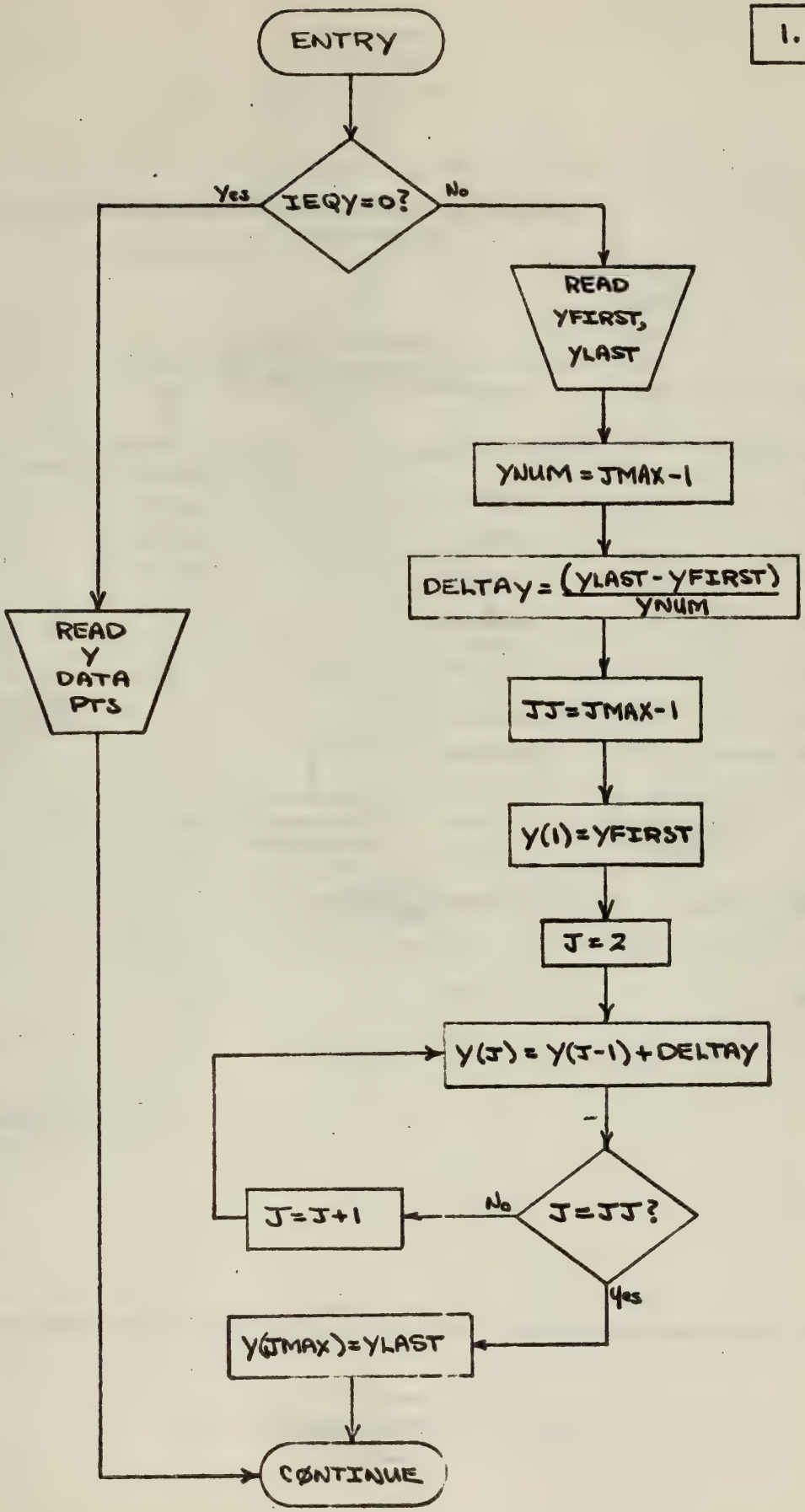



S/R CONVER CONT'D
1
C ≥ 0?
No
Yes
XINT = A(J) - C
A(J) ≥ 0?
No
Yes
DXDT = 180.0 + A(J)
X = 180.0 - C
XINT = DXDT + X
A(J) = (111415.10 * COSF(Y) - 94.55 * COSF(3.0*Y) - .12 * COSF(5.0*Y) * XINT) / .9144
J = J+1
No
J=N?
Yes
M>0?
Yes
No
PRINT K, A(K), B(K)
RETURN

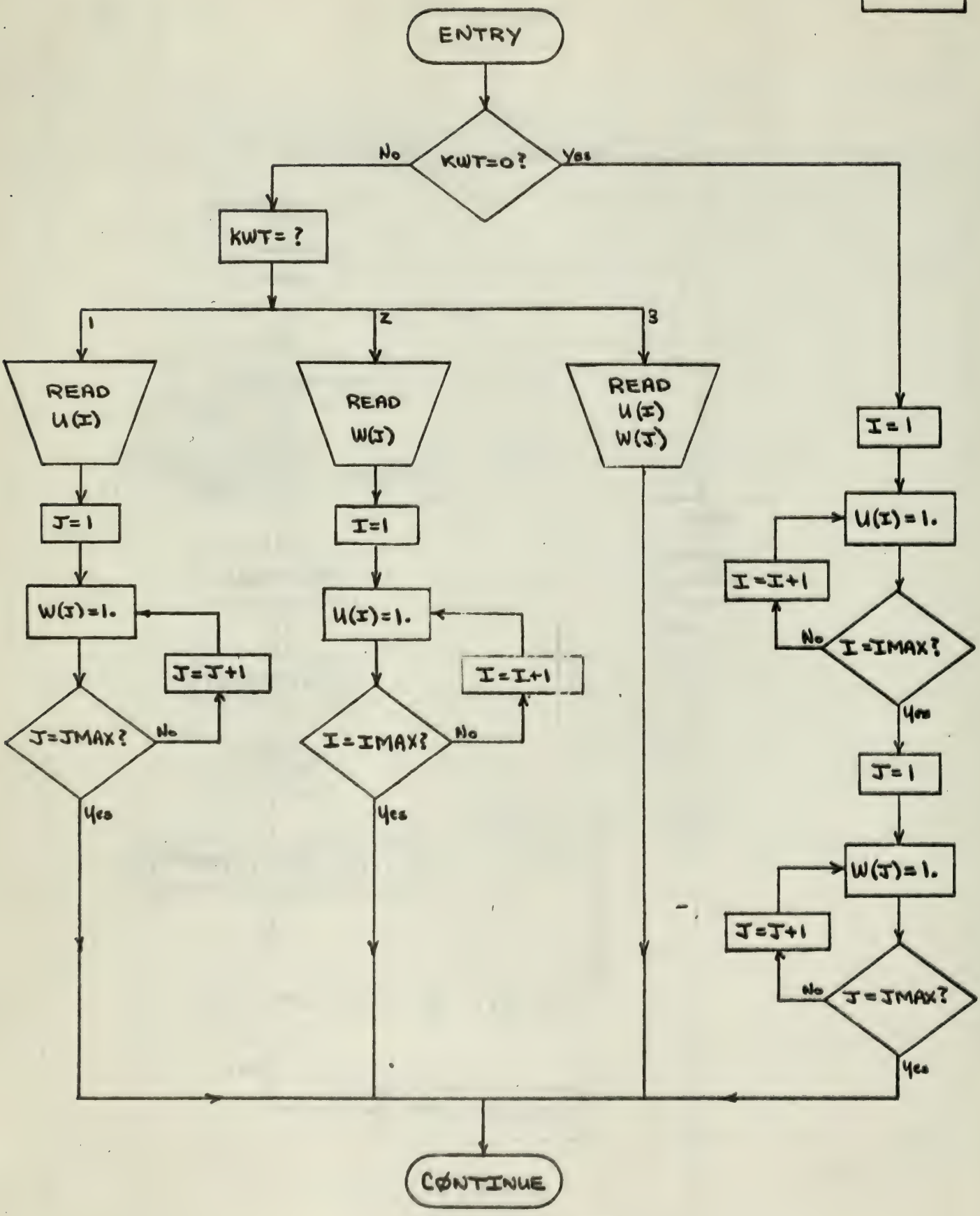
PROGRAM SURFIT



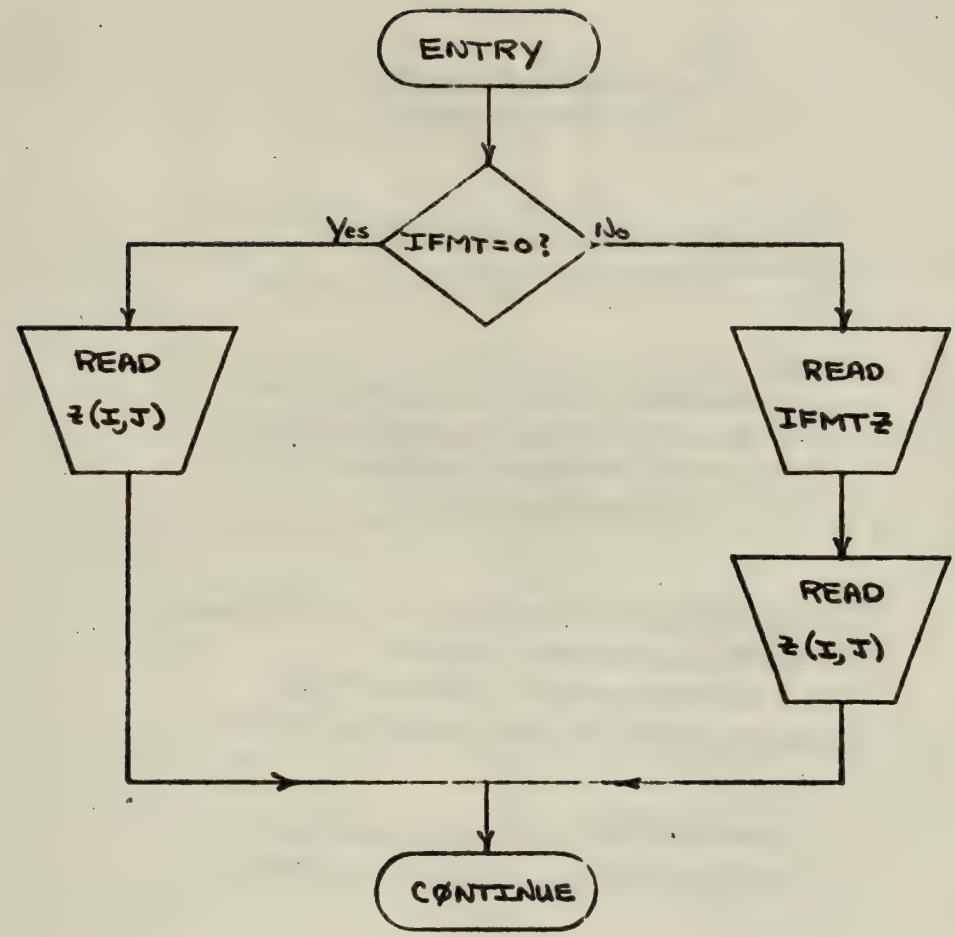

ENTRY
IEQX = 0?
Yes
No
READ X DATA PTS
READ XFIRST, XLAST
XNUM = IMAX-1
DELTAX = (XLAST - XFIRST) / XNUM
II = IMAX - 1
X(1) = XFIRST
I = 2
X(I) = X(I-1) + DELTAX
I = II?
No
I = I+1
Yes
X(IMAX) = XLAST
CONTINUE


ENTRY
IEQY=0?
Yes
No
READ Y DATA PTS
READ YFIRST, YLAST
YNUM = JMAX-1
DELTAY = (YLAST - YFIRST) / YNUM
JJ=JMAX-1
Y(1)=YFIRST
J=2
Y(J) = Y(J-1) + DELTAY
J=JJ?
No
J=J+1
Yes
Y(JMAX)=YLAST
CONTINUE

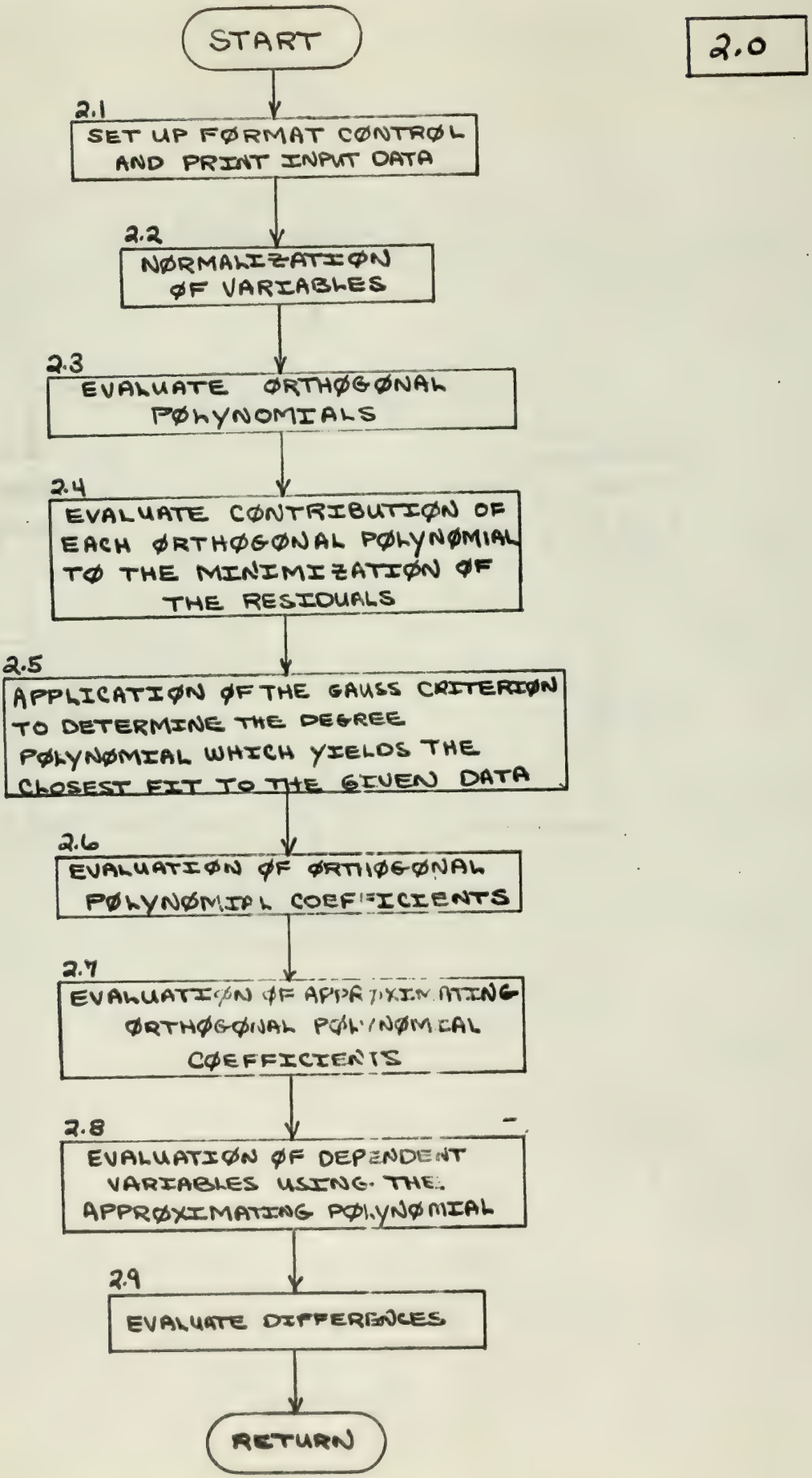
1.3


ENTRY
KWT=0?
No
Yes
KWT=?
1
2
3
READ U(I)
READ W(J)
READ U(I) W(J)
J=1
I=1
I=1
W(J)=1.
U(I)=1.
U(I)=1.
J=J+1
I=I+1
I=I+1
J=JMAX?
No
I=IMAX?
No
Yes
Yes
I=IMAX?
No
Yes
J=1
W(J)=1.
J=J+1
J=JMAX?
No
Yes
CONTINUE

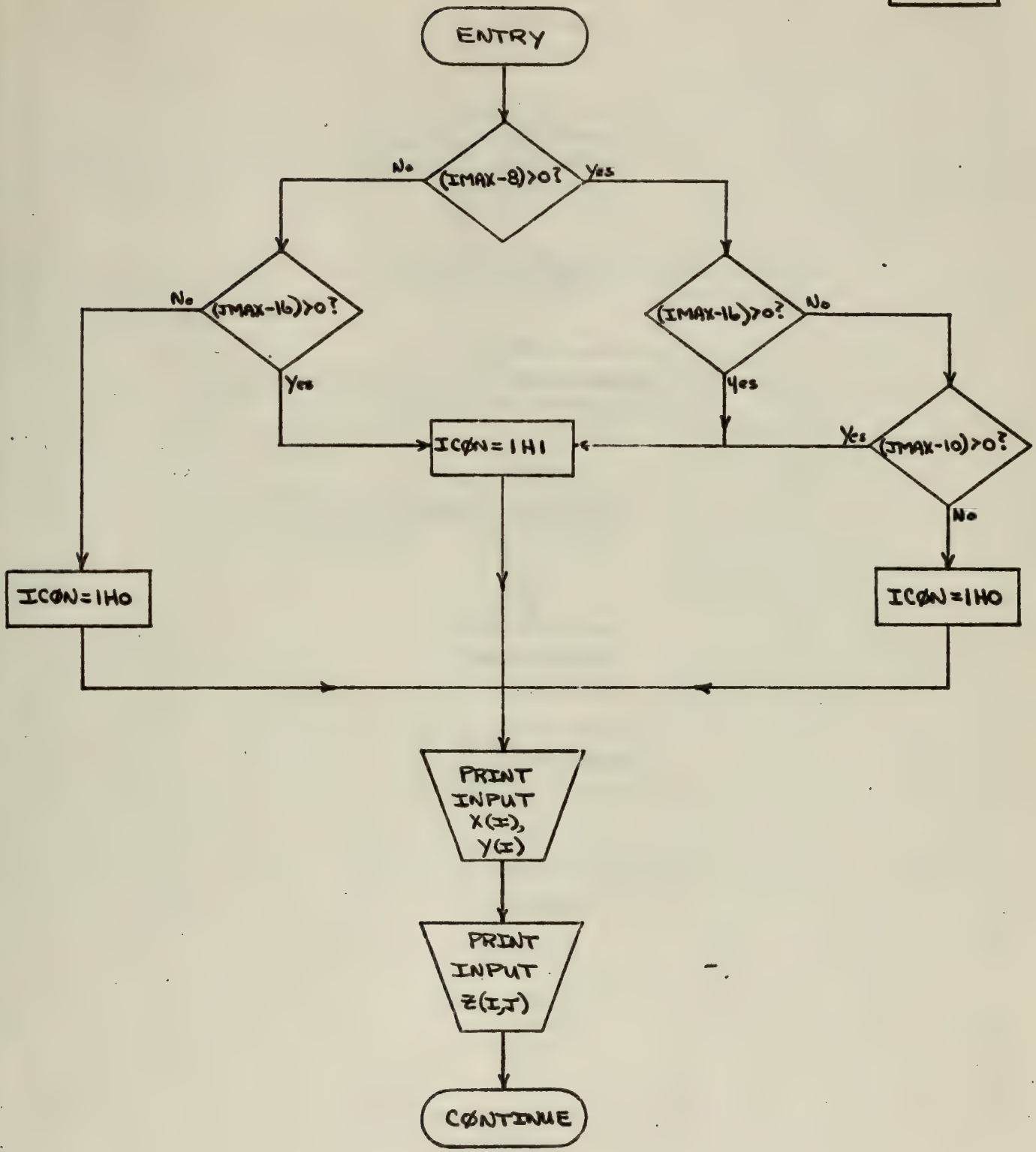
1.4

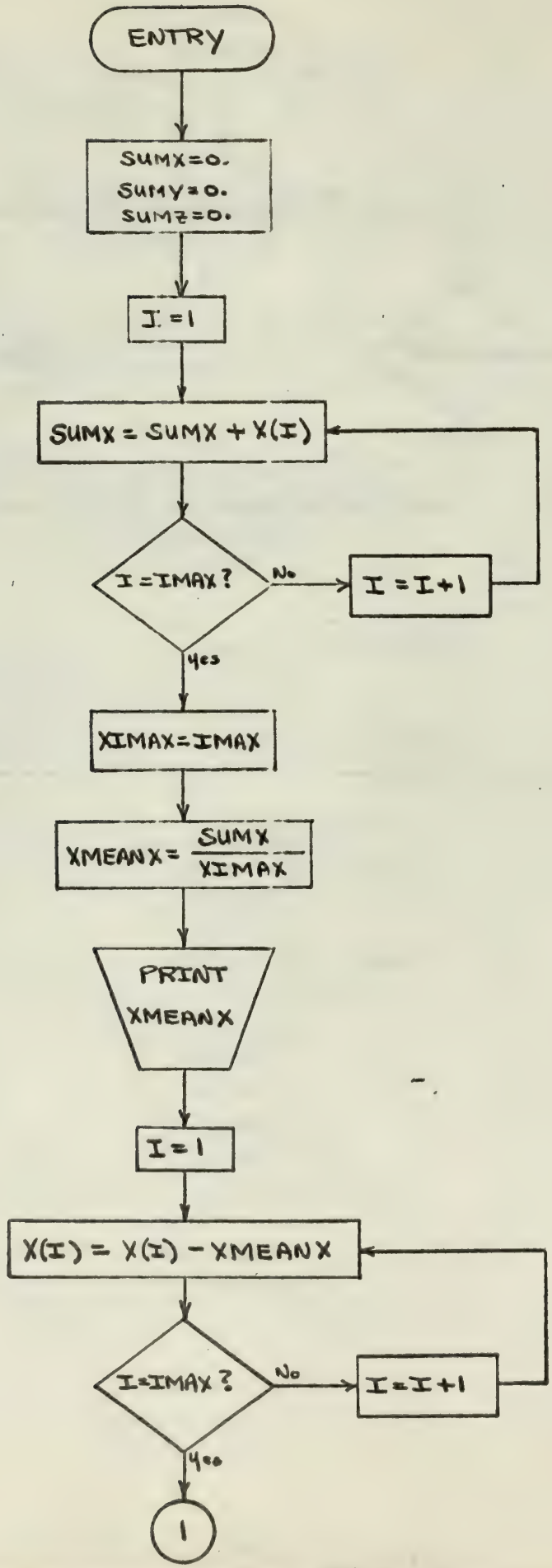


# SUBROUTINE SURFACE

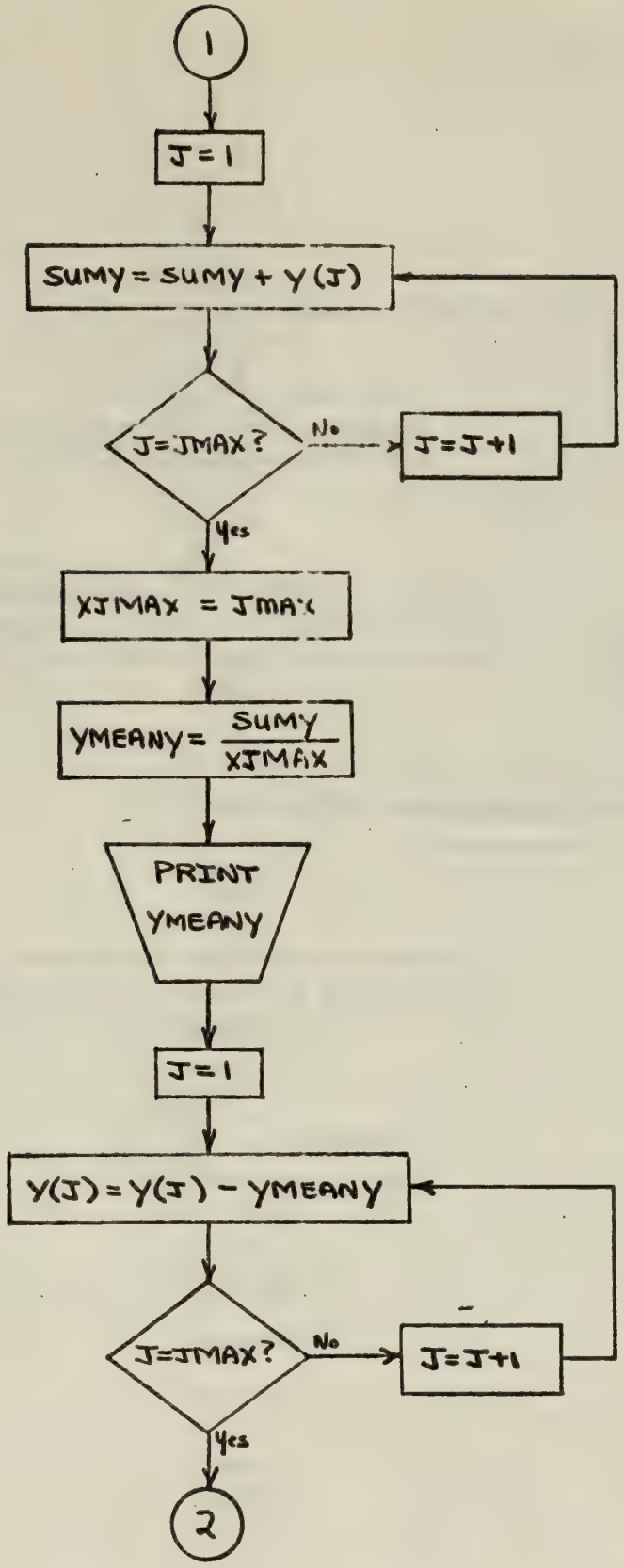


2.1

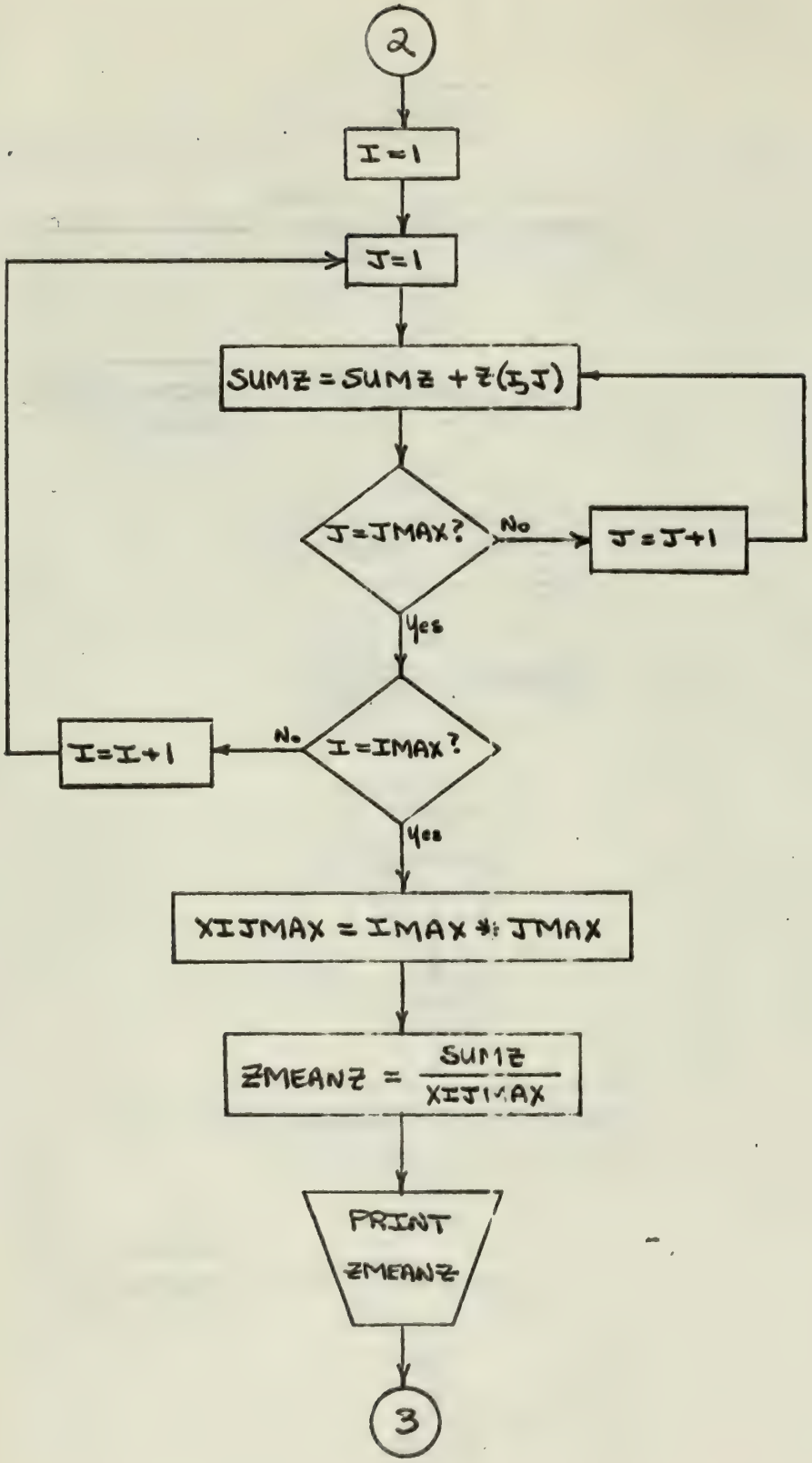

ENTRY
(IMAX-8)>0?
No
Yes
(JMAX-16)>0?
No
Yes
(IMAX-16)>0?
Yes
No
(JMAX-10)>0?
Yes
No
ICØN=1H1
ICØN=1HO
ICØN=1HO
PRINT INPUT X(I), Y(I)
PRINT INPUT Z(I,J)
CØNTINUE

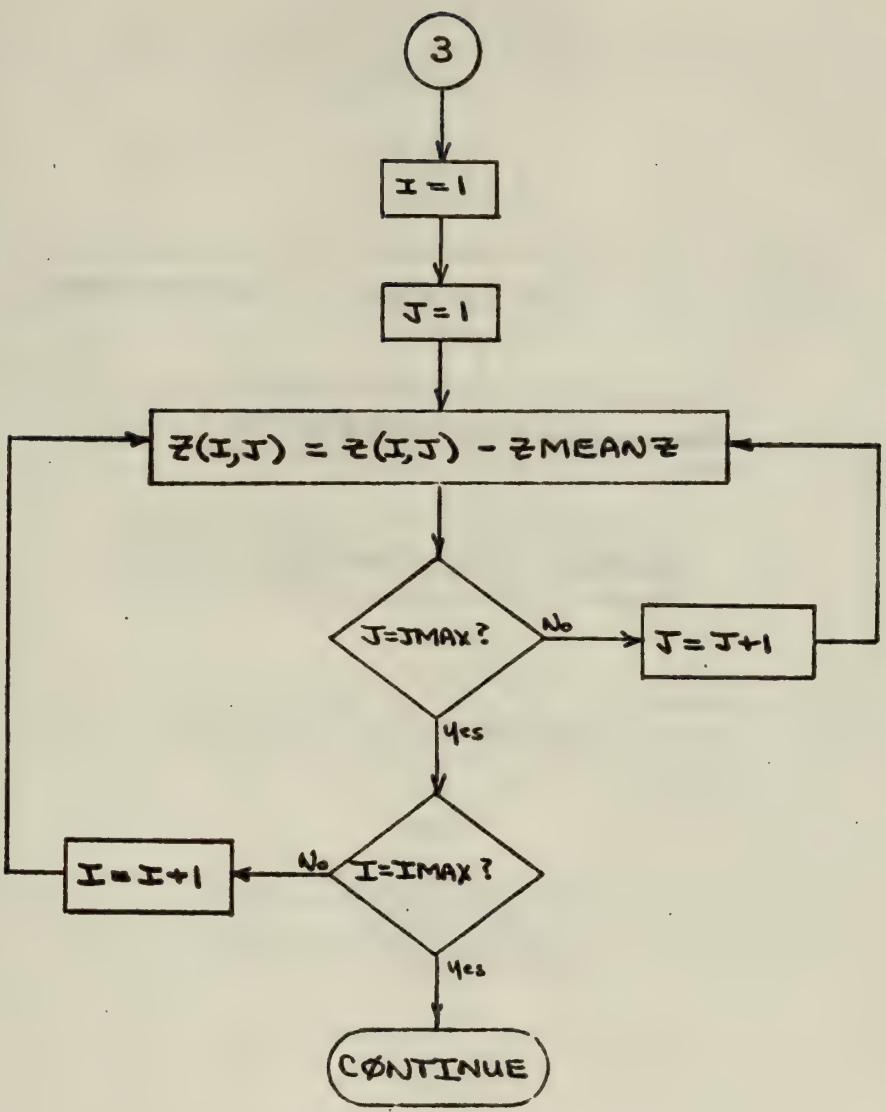

ENTRY
SUMX=0.
SUMY=0.
SUMZ=0.
I=1
SUMX = SUMX + X(I)
I=IMAX?
No
I=I+1
Yes
XIMAX=IMAX
XMEANX = SUMX / XIMAX
PRINT
XMEANX
I=1
X(I) = X(I) - XMEANX
I=IMAX?
No
I=I+1
Yes
1

2.2 (CONT'D)

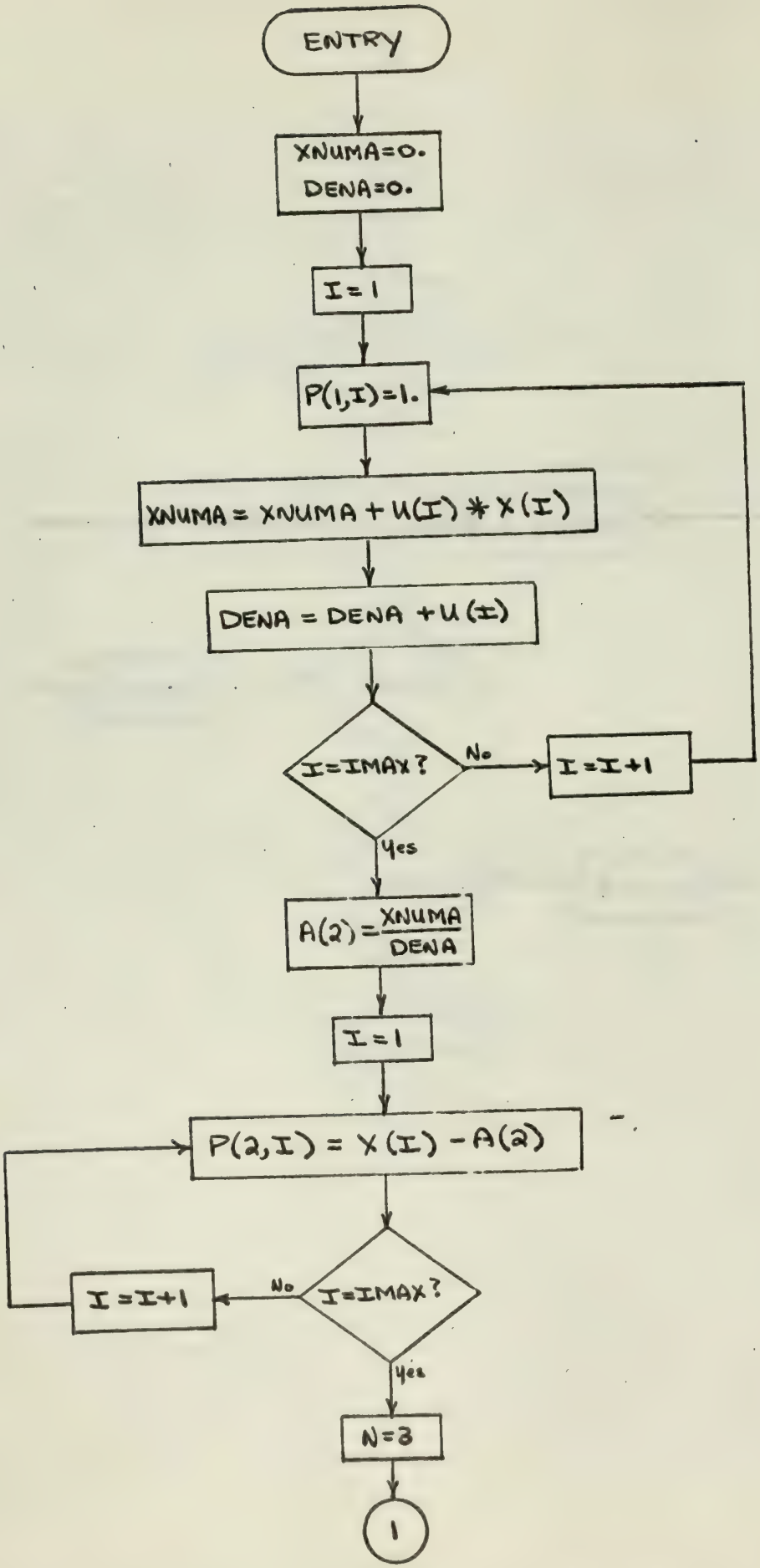

1
J=1
SUMY = SUMY + Y(J)
J=JMAX?
No
J=J+1
Yes
XJMAX = JMAX
YMEANY = SUMY / XJMAX
PRINT YMEANY
J=1
Y(J) = Y(J) - YMEANY
J=JMAX?
No
J=J+1
Yes
2

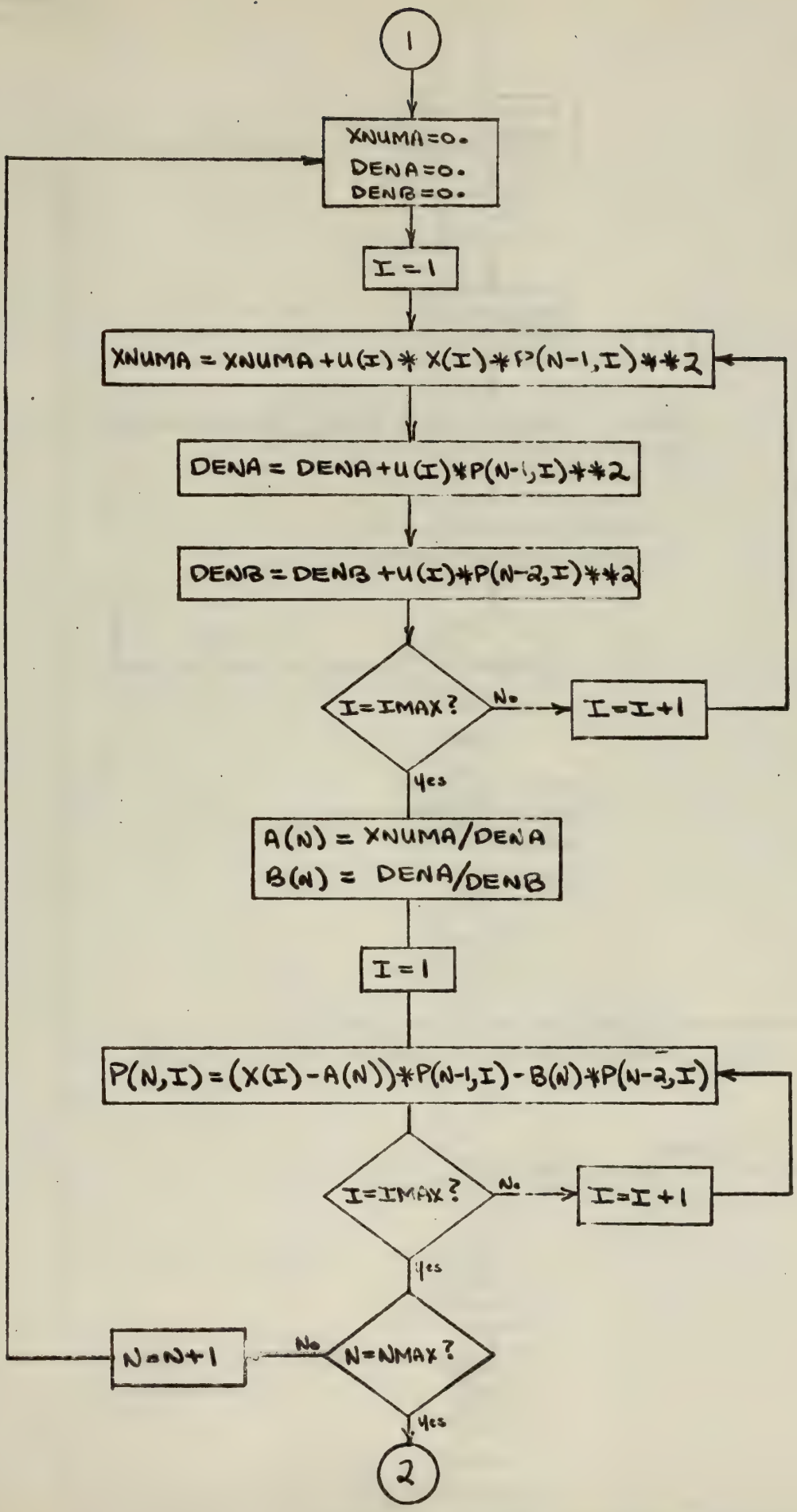
2.2
(CONT'D)


2
I=1
J=1
SUMZ = SUMZ + Z(I,J)
J=JMAX?
No
J = J+1
Yes
I=IMAX?
No
I=I+1
Yes
XIJMAX = IMAX * JMAX
ZMEANZ = SUMZ / XIJMAX
PRINT
ZMEANZ
3

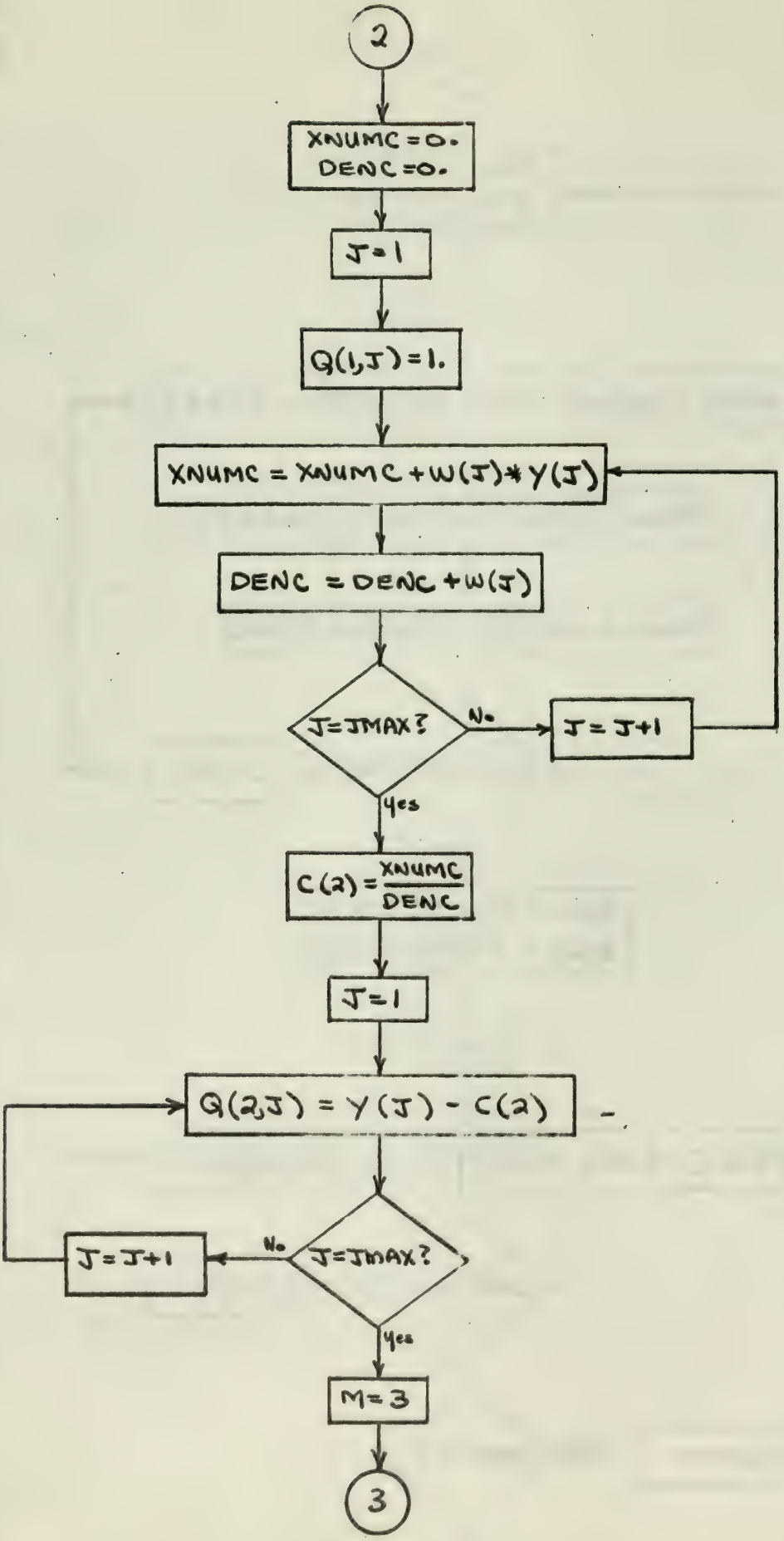

3
I = 1
J = 1
Z(I,J) = Z(I,J) - ZMEANZ
J=JMAX?
No
J = J+1
Yes
I=IMAX?
No
I = I+1
Yes
CONTINUE

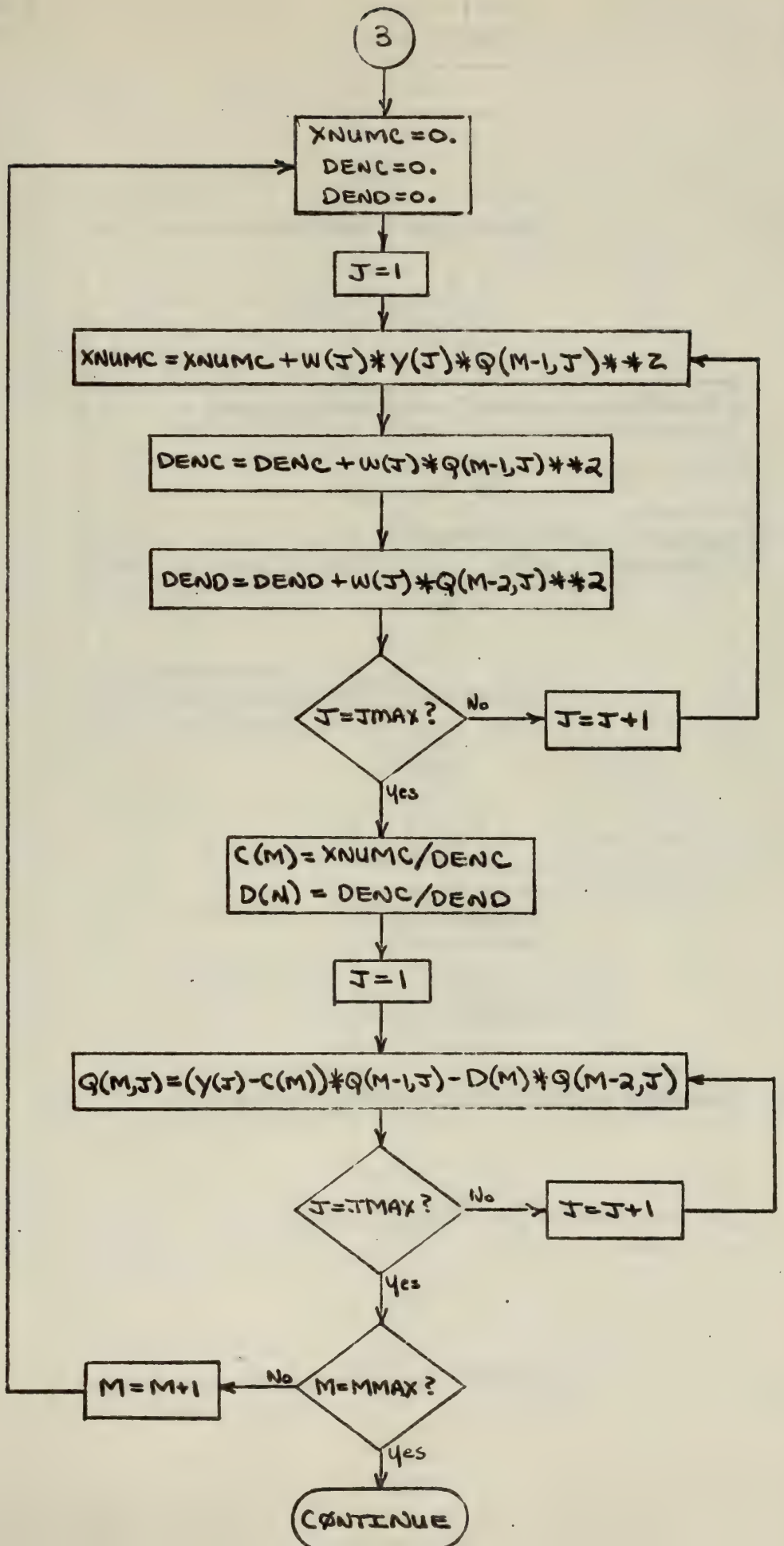
2.3

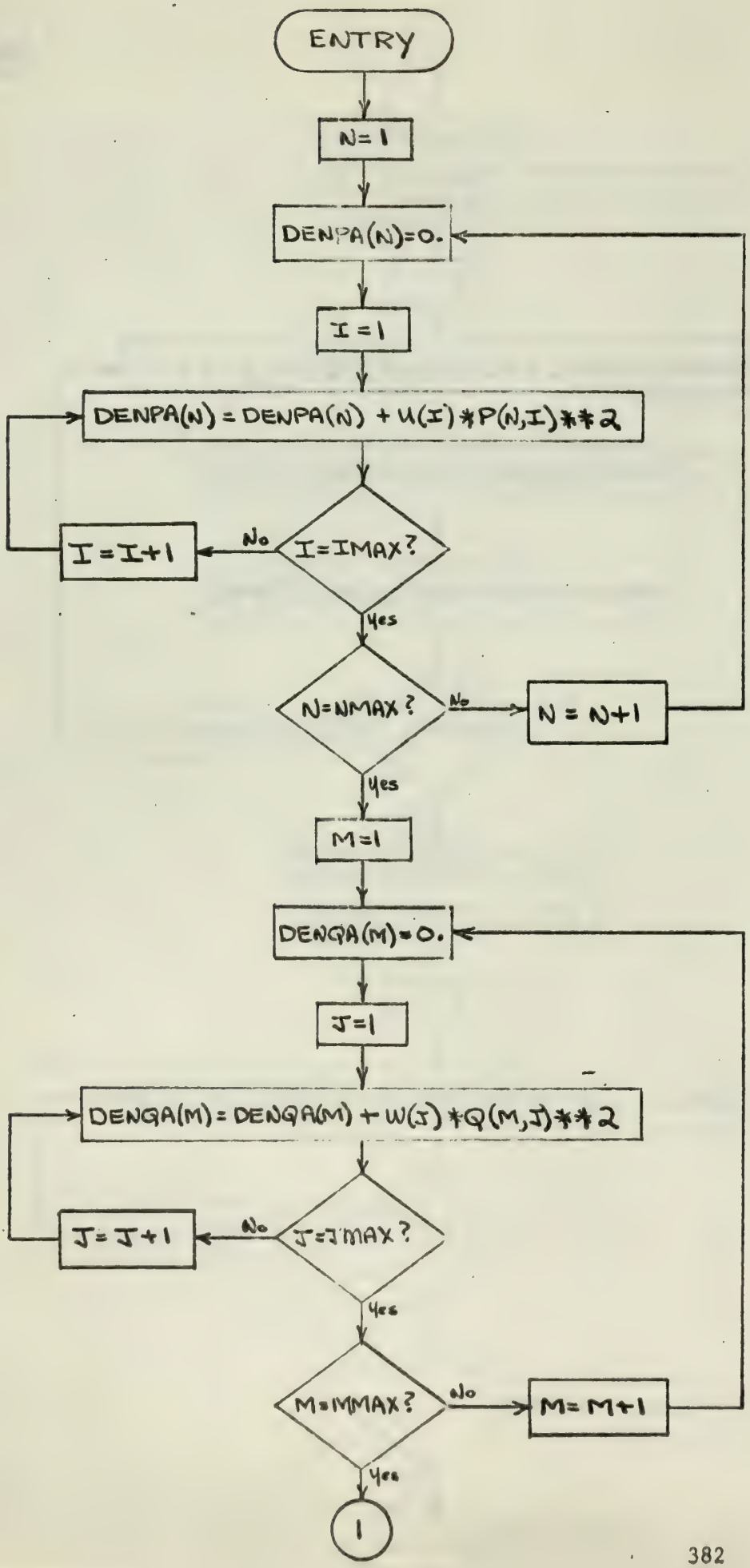

ENTRY
XNUMA=0.
DENA=0.
I=1
P(1,I)=1.
XNUMA = XNUMA + U(I) * X(I)
DENA = DENA + U(I)
I=IMAX?
No
I=I+1
Yes
A(2)=XNUMA/DENA
I=1
P(2,I) = X(I) - A(2)
I=IMAX?
No
I=I+1
Yes
N=3
1

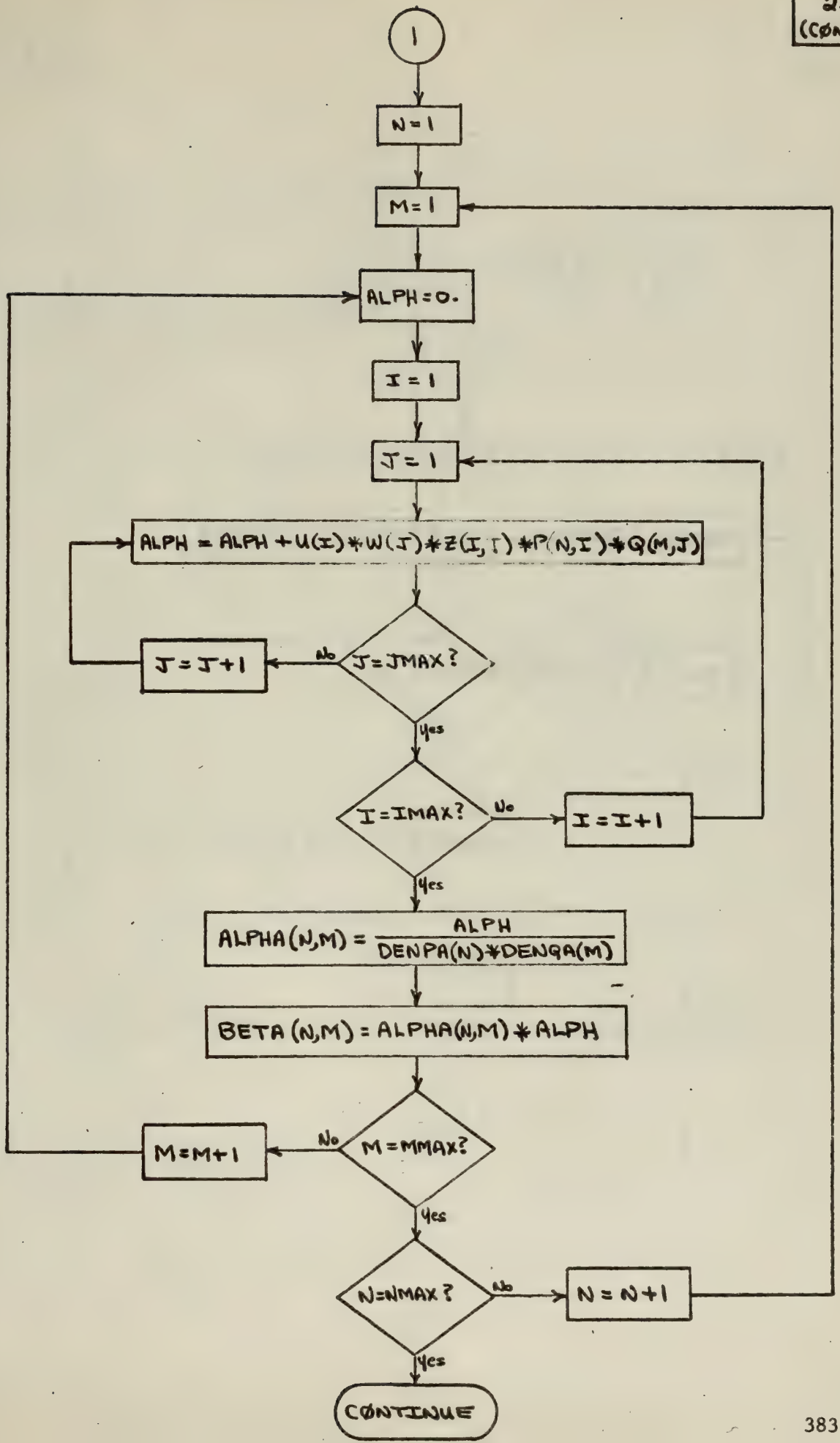

1
XNUMA=0.
DENA=0.
DENB=0.
I=1
XNUMA = XNUMA+U(I)*X(I)*P(N-1,I)**2
DENA = DENA+U(I)*P(N-1,I)**2
DENB = DENB+U(I)*P(N-2,I)**2
I=IMAX?
No
I=I+1
Yes
A(N) = XNUMA/DENA
B(N) = DENA/DENB
I=1
P(N,I) = (X(I)-A(N))*P(N-1,I)-B(N)*P(N-2,I)
I=IMAX?
No
I=I+1
Yes
N=NMAX?
No
N=N+1
Yes
2

2.3 (CONT'D)

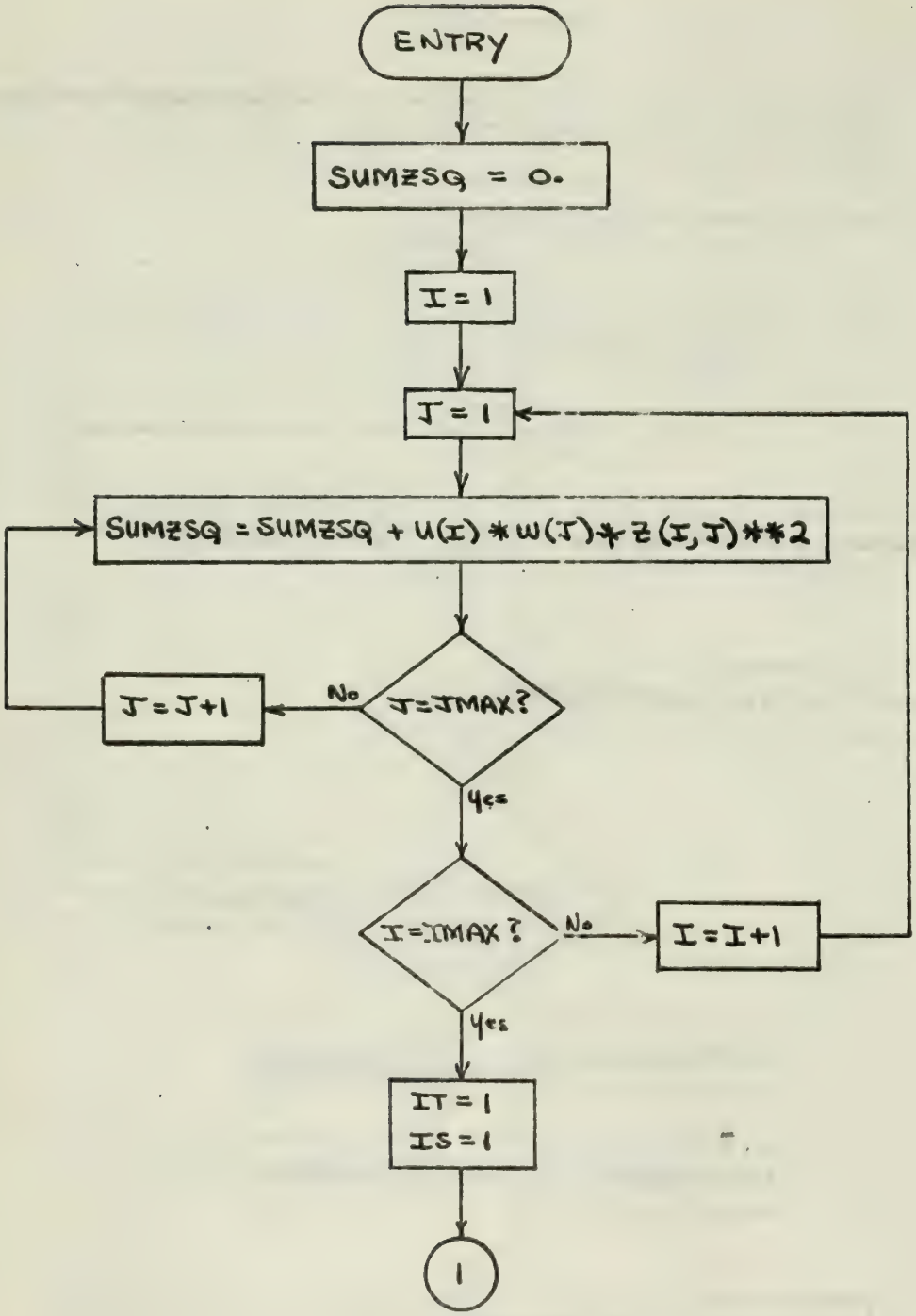

2
XNUMC = 0.
DENC = 0.
J = 1
Q(1,J) = 1.
XNUMC = XNUMC + W(J) * Y(J)
DENC = DENC + W(J)
J = JMAX?
No
J = J+1
Yes
C(2) = XNUMC / DENC
J = 1
Q(2,J) = Y(J) - C(2)
J = JMAX?
No
J = J+1
Yes
M = 3
3


3
XNUMC=0.
DENC=0.
DEND=0.
J=1
XNUMC=XNUMC+W(J)*Y(J)*Q(M-1,J)**2
DENC=DENC+W(J)*Q(M-1,J)**2
DEND=DEND+W(J)*Q(M-2,J)**2
J=JMAX?
No
J=J+1
Yes
C(M)=XNUMC/DENC
D(N)=DENC/DEND
J=1
Q(M,J)=(Y(J)-C(M))*Q(M-1,J)-D(M)*Q(M-2,J)
J=JMAX?
No
J=J+1
Yes
M=MMAX?
No
M=M+1
Yes
CONTINUE

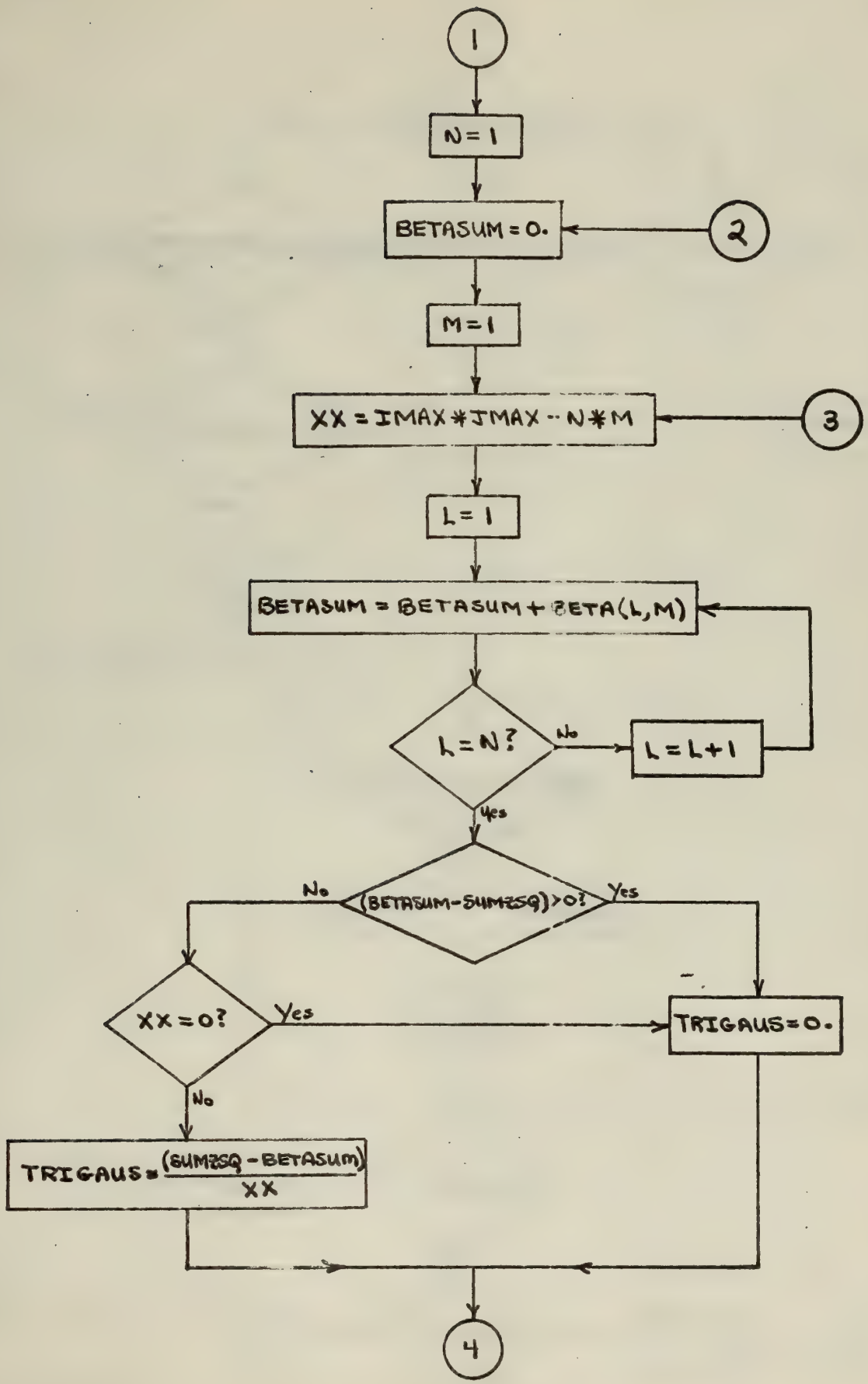

ENTRY
N=1
DENPA(N)=0.
I=1
DENPA(N) = DENPA(N) + U(I)*P(N,I)**2
I=I+1
No
I=IMAX?
Yes
N=NMAX?
No
N = N+1
Yes
M=1
DENQA(M)=0.
J=1
DENQA(M) = DENQA(M) + W(J)*Q(M,J)**2
J=J+1
No
J=JMAX?
Yes
M=MMAX?
No
M=M+1
Yes
1

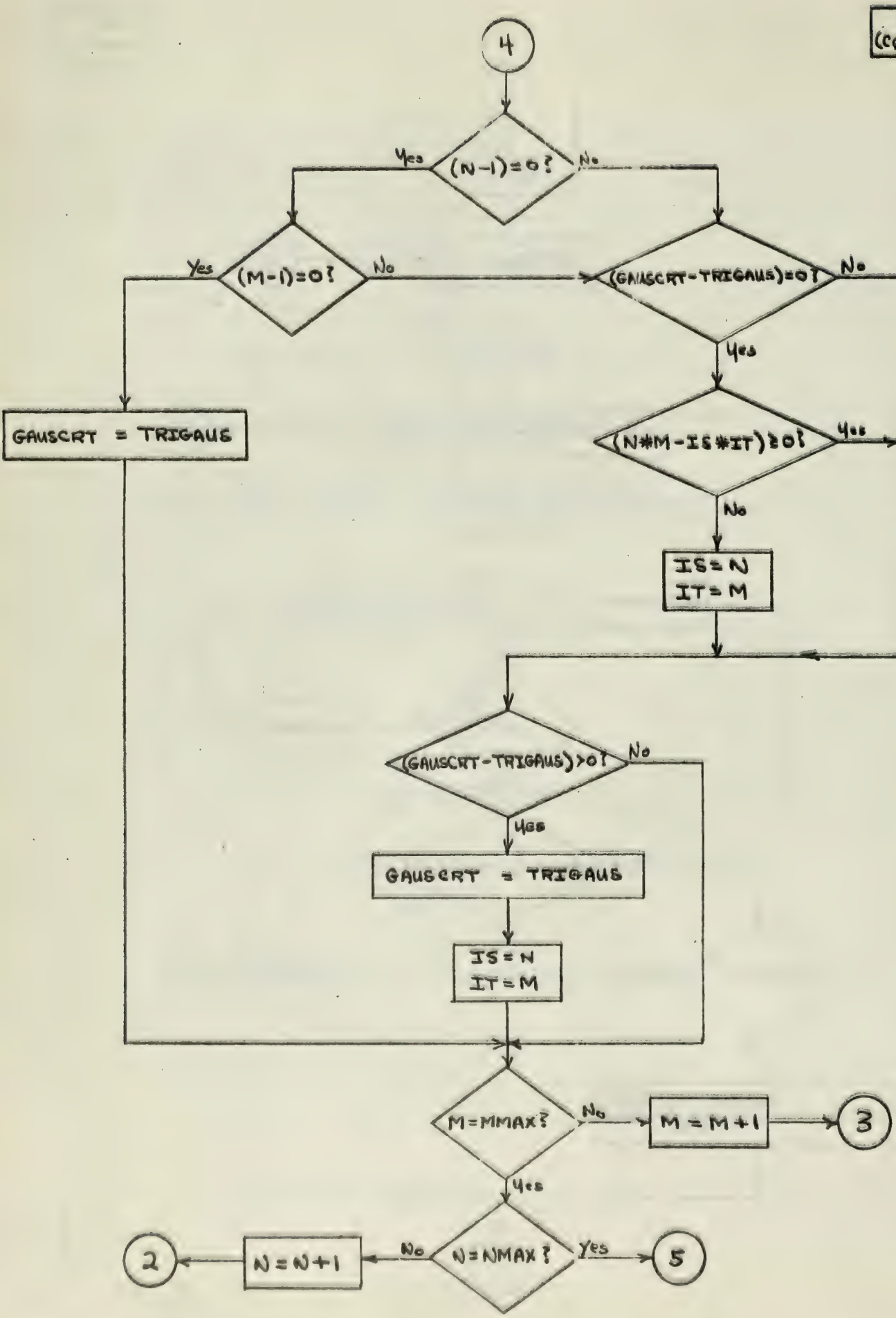

1
N=1
M=1
ALPH=0.
I=1
J=1
ALPH = ALPH + U(I)*W(J)*Z(I,J)*P(N,I)*Q(M,J)
J=J+1
No
J=JMAX?
Yes
I=IMAX?
No
I=I+1
Yes
ALPHA(N,M) = ALPH / DENPA(N)*DENQA(M)
BETA(N,M) = ALPHA(N,M)*ALPH
M=M+1
No
M=MMAX?
Yes
N=NMAX?
No
N=N+1
Yes
CONTINUE

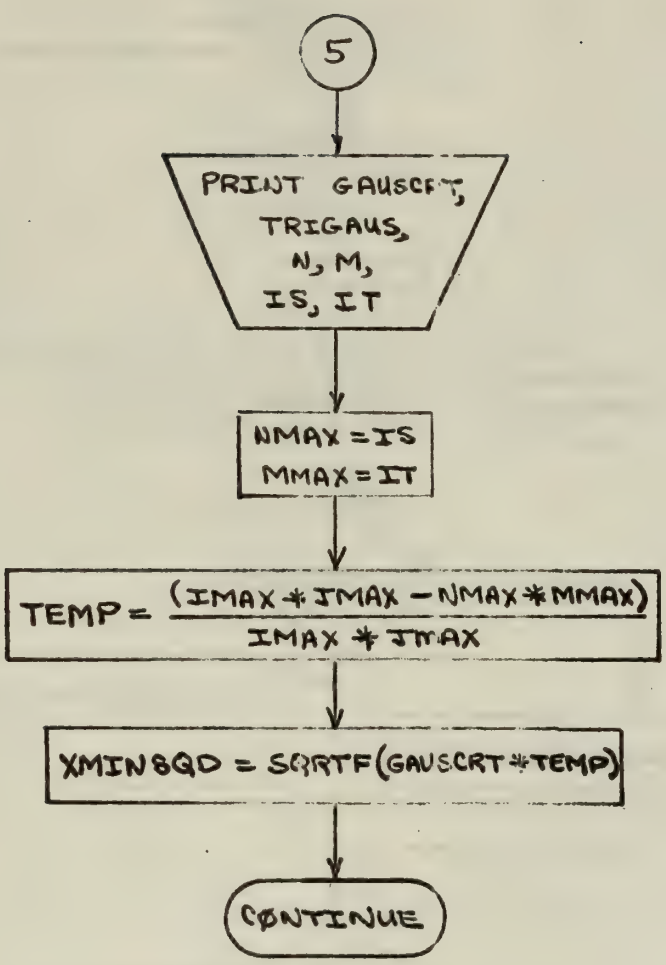
2.5

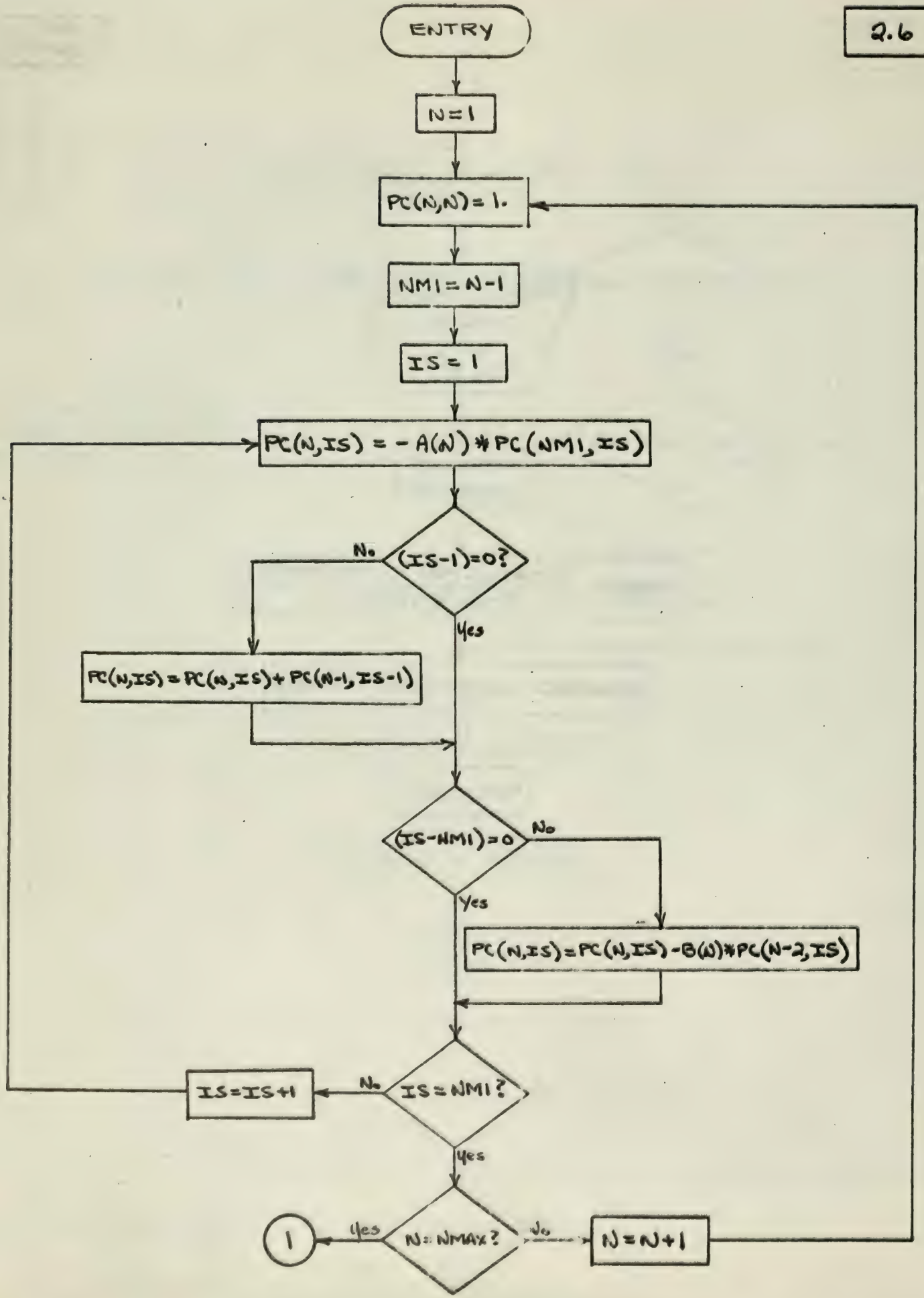

ENTRY
SUMZSQ = 0.
I = 1
J = 1
SUMZSQ = SUMZSQ + U(I) * W(J) * Z(I,J) **2
J=JMAX?
No
J = J+1
Yes
I=IMAX?
No
I = I+1
Yes
IT = 1
IS = 1
1

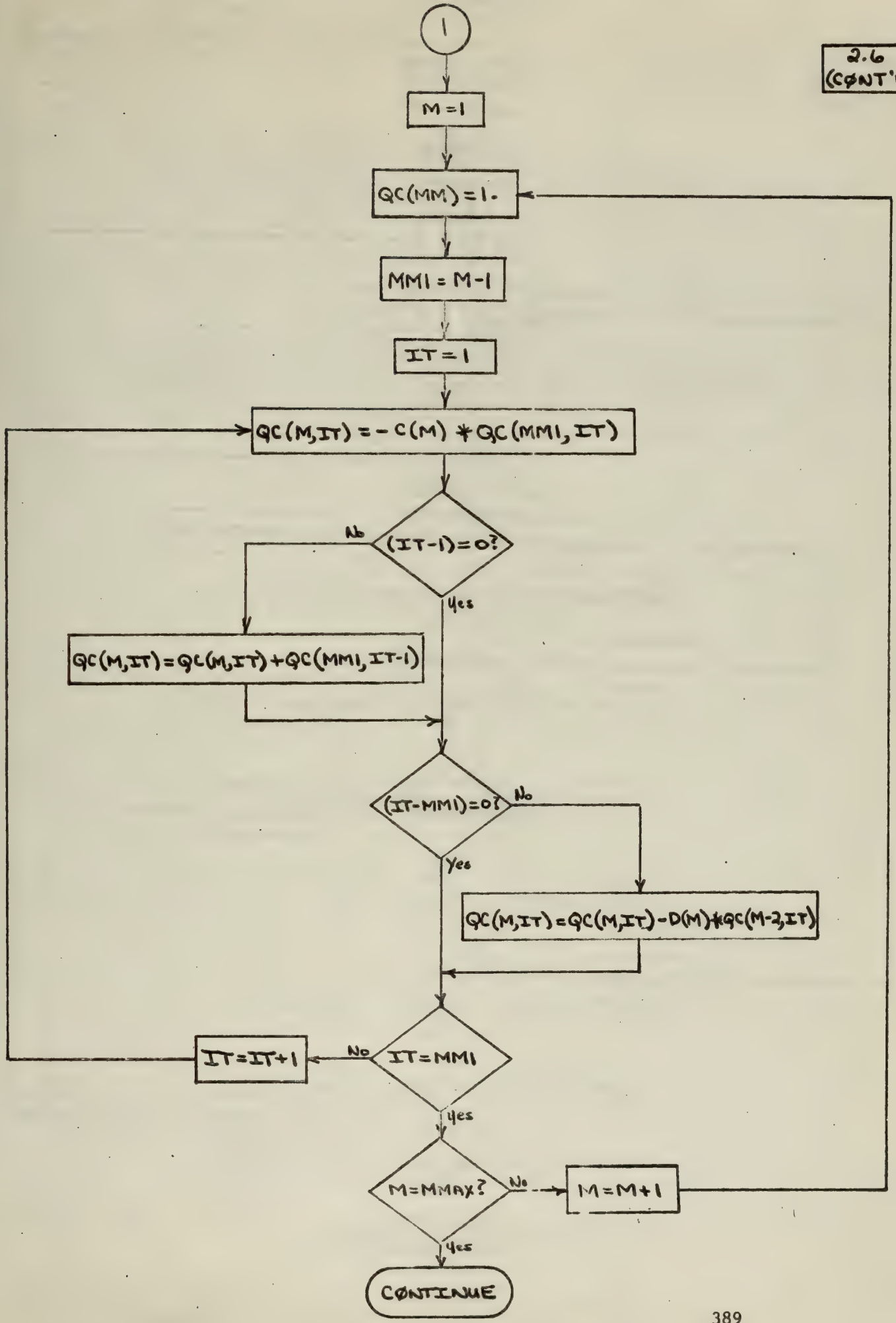
2.5 (CONT'D)

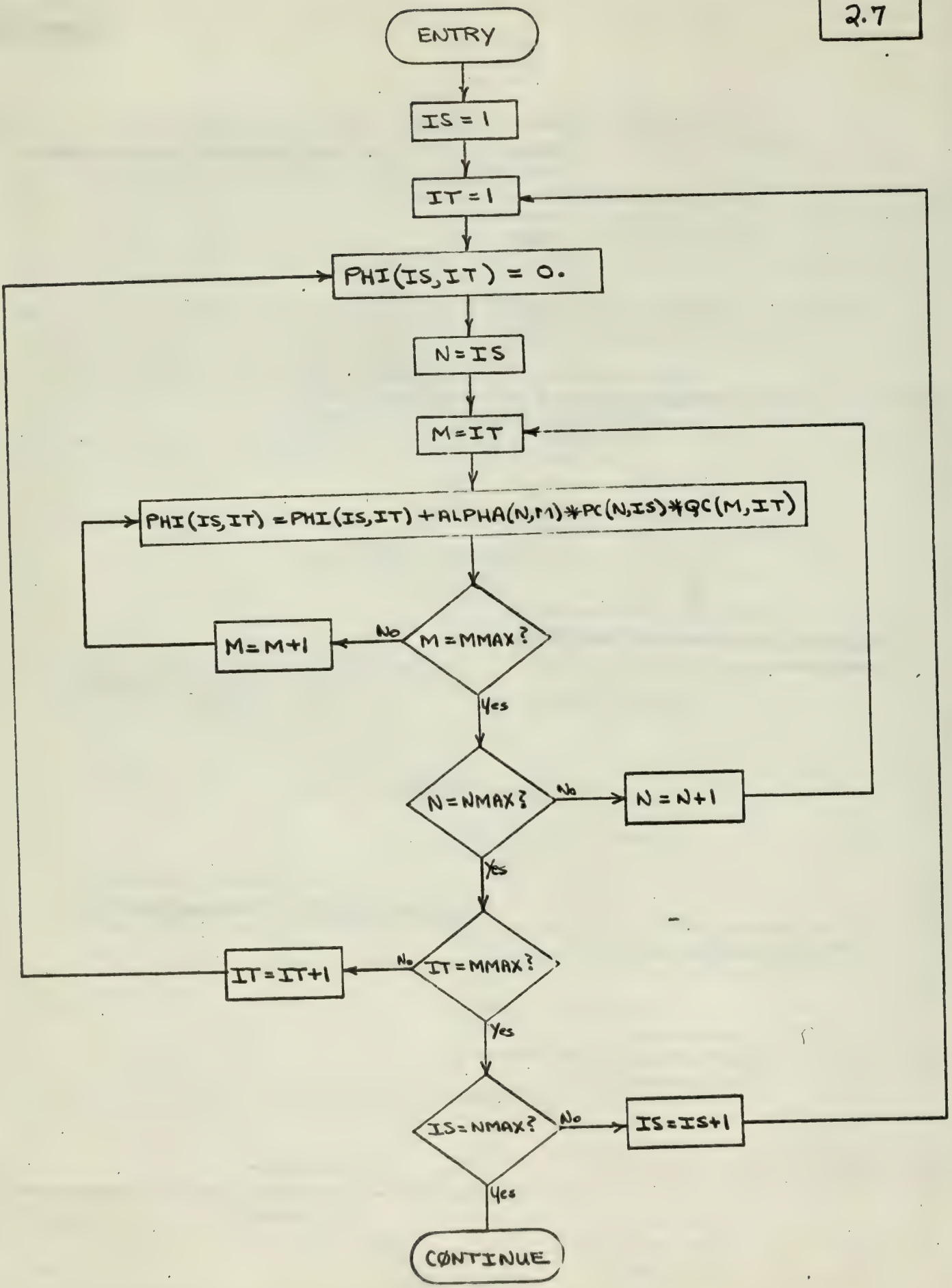

1
N=1
BETASUM=0.
2
M=1
XX=IMAX*JMAX-N*M
3
L=1
BETASUM=BETASUM+BETA(L,M)
L=N?
No
L=L+1
Yes
(BETASUM-SUMZSQ)>0?
No
Yes
XX=0?
Yes
TRIGAUS=0.
No
TRIGAUS=(SUMZSQ-BETASUM)/XX
4

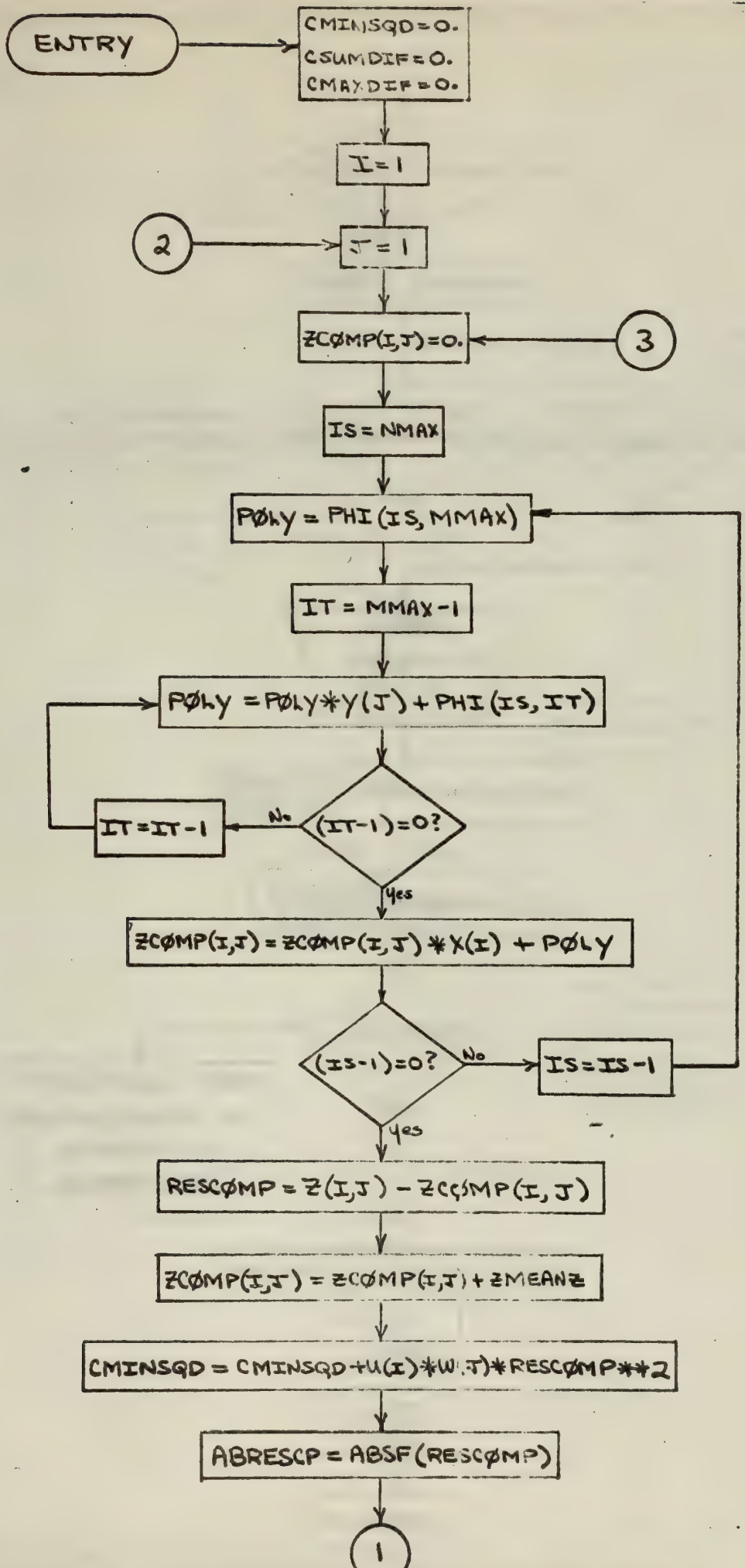

4
(N-1)=0?
Yes
No
(M-1)=0?
Yes
No
(GAUSCRT-TRIGAUS)=0?
No
Yes
GAUSCRT = TRIGAUS
(N*M-IS*IT)≥0?
Yes
No
IS=N
IT=M
(GAUSCRT-TRIGAUS)>0?
No
Yes
GAUSCRT = TRIGAUS
IS=N
IT=M
M=MMAX?
No
M=M+1
3
Yes
N=NMAX?
No
N=N+1
2
Yes
5

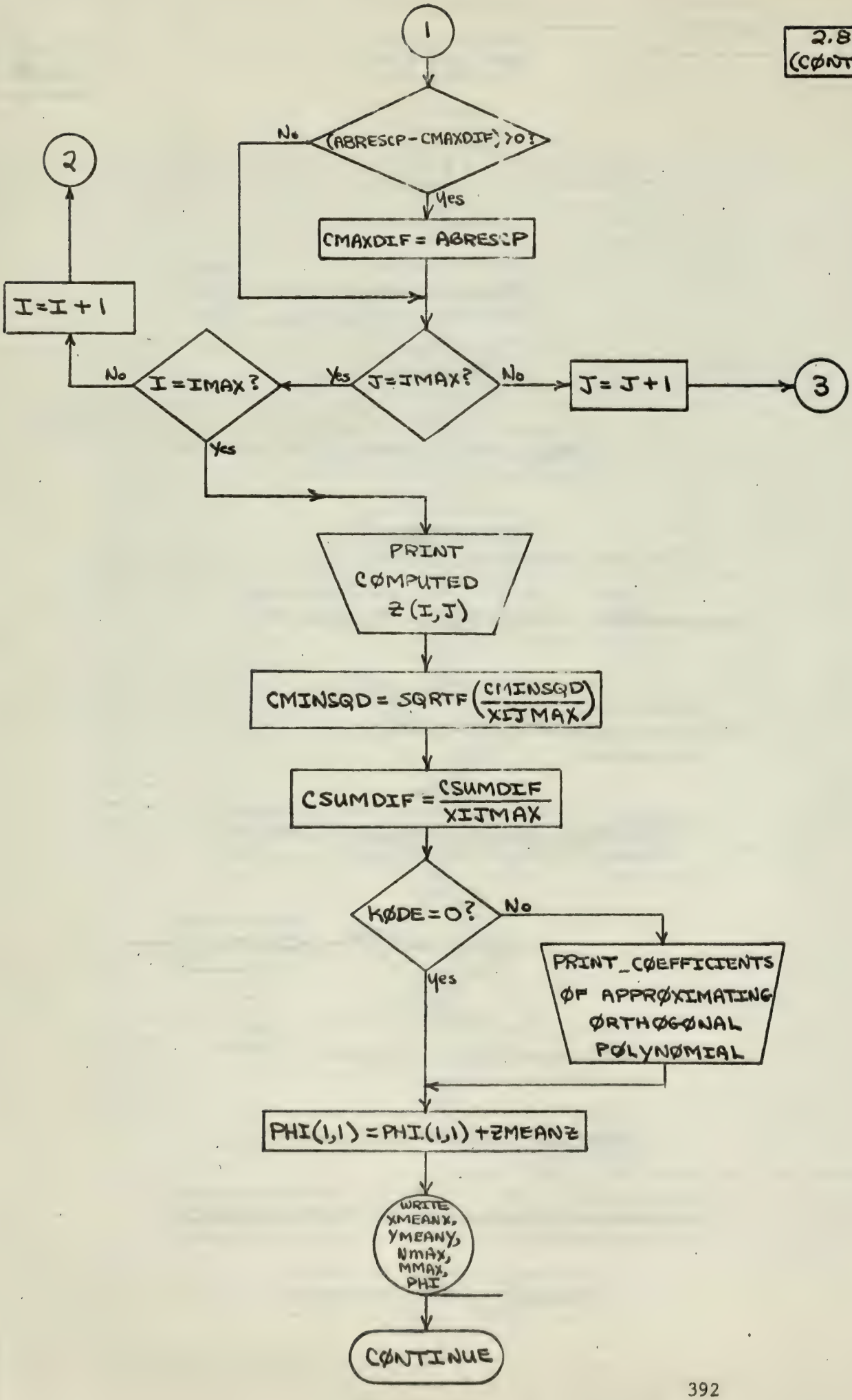

5
PRINT GAUSCRT,
TRIGAUS,
N, M,
IS, IT
NMAX = IS
MMAX = IT
TEMP = (IMAX * JMAX − NMAX * MMAX) / IMAX * JMAX
XMINSQD = SQRTF(GAUSCRT * TEMP)
CONTINUE

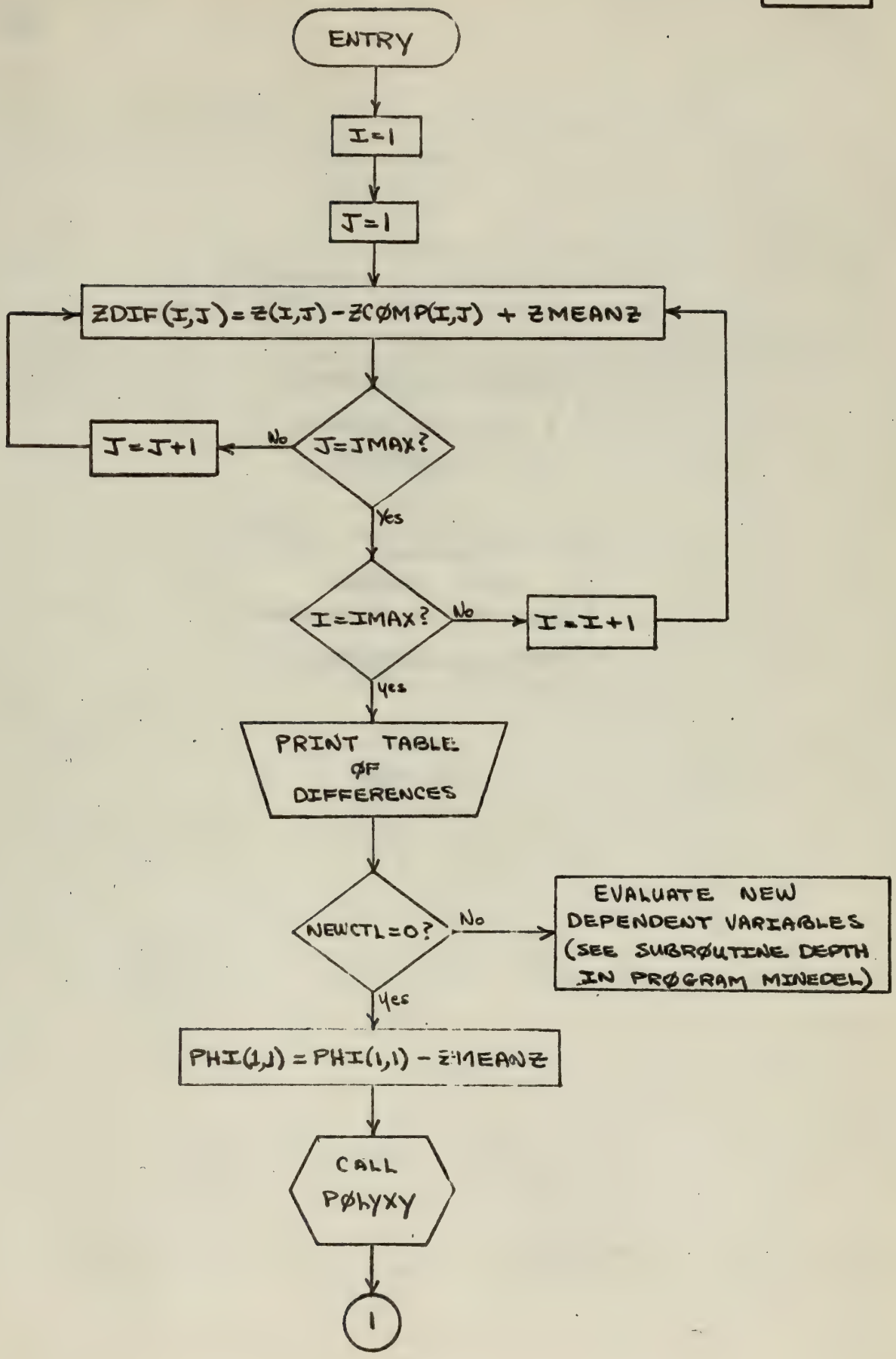

ENTRY
N=1
PC(N,N)=1.
NM1=N-1
IS=1
PC(N,IS)=-A(N)*PC(NM1,IS)
(IS-1)=0?
No
Yes
PC(N,IS)=PC(N,IS)+PC(N-1,IS-1)
(IS-NM1)=0
No
Yes
PC(N,IS)=PC(N,IS)-B(N)*PC(N-2,IS)
IS=NM1?
No
IS=IS+1
Yes
N=NMAX?
Yes
1
No
N=N+1

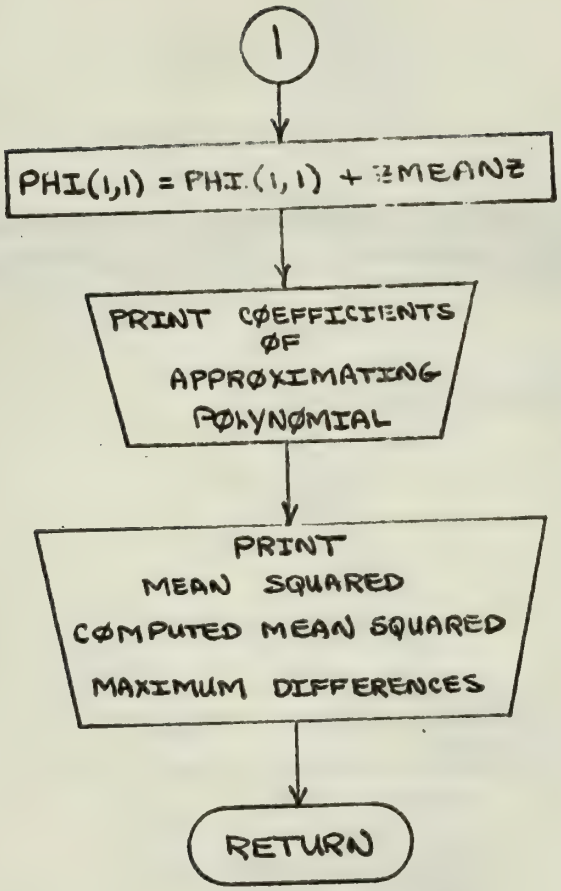

1
M=1
QC(MM)=1.
MM1=M-1
IT=1
QC(M,IT)=-C(M)*QC(MM1,IT)
(IT-1)=0?
No
Yes
QC(M,IT)=QC(M,IT)+QC(MM1,IT-1)
(IT-MM1)=0?
No
Yes
QC(M,IT)=QC(M,IT)-D(M)*QC(M-2,IT)
IT=MM1
No
IT=IT+1
Yes
M=MMAX?
No
M=M+1
Yes
CONTINUE


ENTRY
IS = 1
IT = 1
PHI(IS,IT) = 0.
N = IS
M = IT
PHI(IS,IT) = PHI(IS,IT) + ALPHA(N,M)*PC(N,IS)*QC(M,IT)
M = MMAX?
No
M = M+1
Yes
N = NMAX?
No
N = N+1
Yes
IT = MMAX?
No
IT = IT+1
Yes
IS = NMAX?
No
IS = IS+1
Yes
CONTINUE

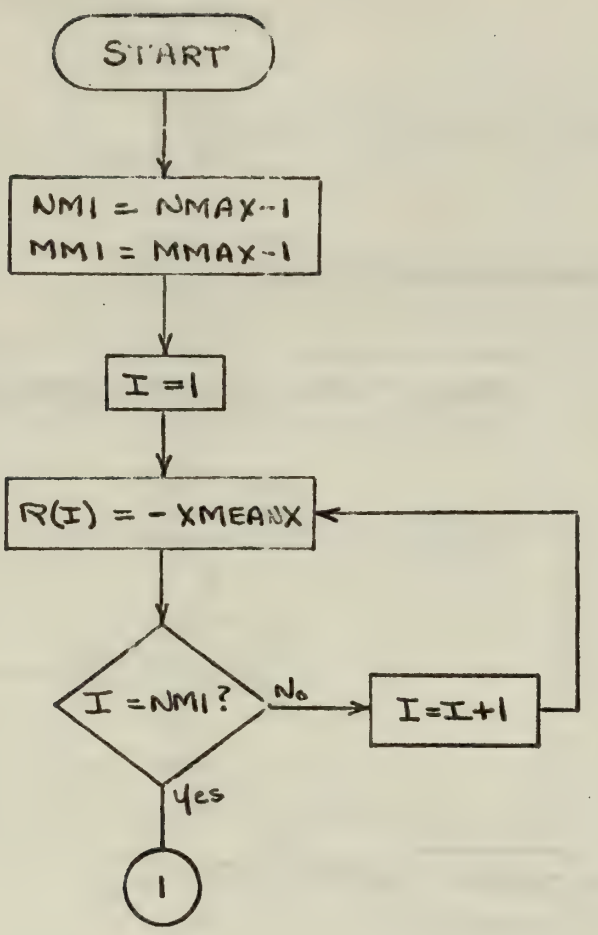

ENTRY
CMINSQD=0.
CSUMDIF=0.
CMAXDIF=0.
I=1
2
J=1
ZCOMP(I,J)=0.
3
IS=NMAX
POLY=PHI(IS,MMAX)
IT=MMAX-1
POLY=POLY*Y(J)+PHI(IS,IT)
(IT-1)=0?
No
IT=IT-1
Yes
ZCOMP(I,J)=ZCOMP(I,J)*X(I)+POLY
(IS-1)=0?
No
IS=IS-1
Yes
RESCOMP=Z(I,J)-ZCOMP(I,J)
ZCOMP(I,J)=ZCOMP(I,J)+ZMEANZ
CMINSQD=CMINSQD+W(I)*W(J)*RESCOMP**2
ABRESCP=ABSF(RESCOMP)
1


1
(ABRESCP-CMAXDIF)>0?
No
Yes
CMAXDIF = ABRESCP
2
I=I+1
No
I=IMAX?
Yes
J=JMAX?
No
J=J+1
3
Yes
PRINT
COMPUTED
Z(I,J)
CMINSQD = SQRTF(CMINSQD/XIJMAX)
CSUMDIF = CSUMDIF/XIJMAX
KODE=0?
No
Yes
PRINT_COEFFICIENTS
OF APPROXIMATING
ORTHOGONAL
POLYNOMIAL
PHI(1,1) = PHI(1,1) + ZMEANZ
WRITE
XMEANX,
YMEANY,
NMAX,
MMAX,
PHI
CONTINUE

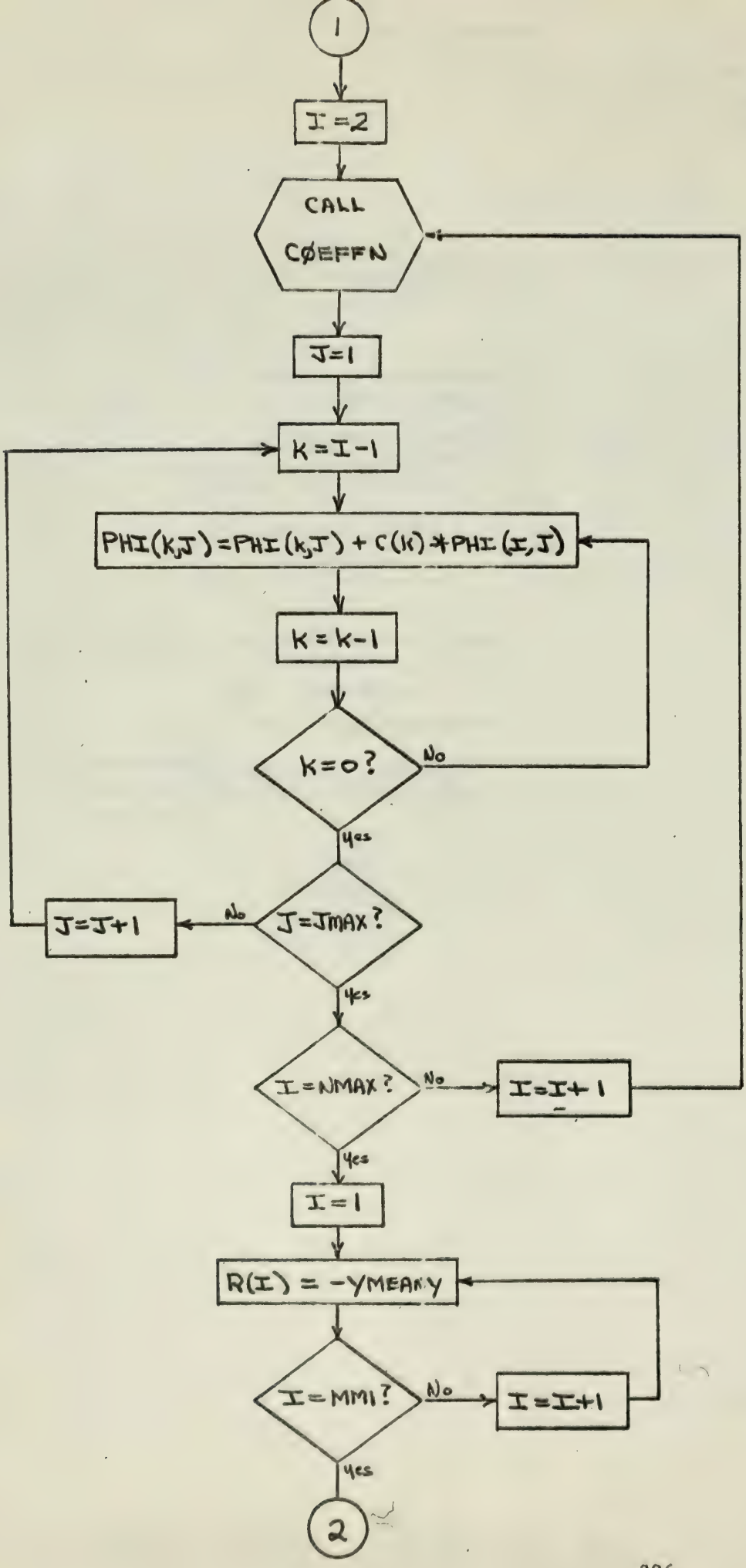

ENTRY
I=1
J=1
ZDIF(I,J) = Z(I,J) - ZCØMP(I,J) + ZMEANZ
J=JMAX?
No
J=J+1
Yes
I=IMAX?
No
I=I+1
Yes
PRINT TABLE ØF DIFFERENCES
NEWCTL=0?
No
EVALUATE NEW DEPENDENT VARIABLES (SEE SUBRØUTINE DEPTH IN PRØGRAM MINEDEL)
Yes
PHI(1,1) = PHI(1,1) - ZMEANZ
CALL PØLYXY
1

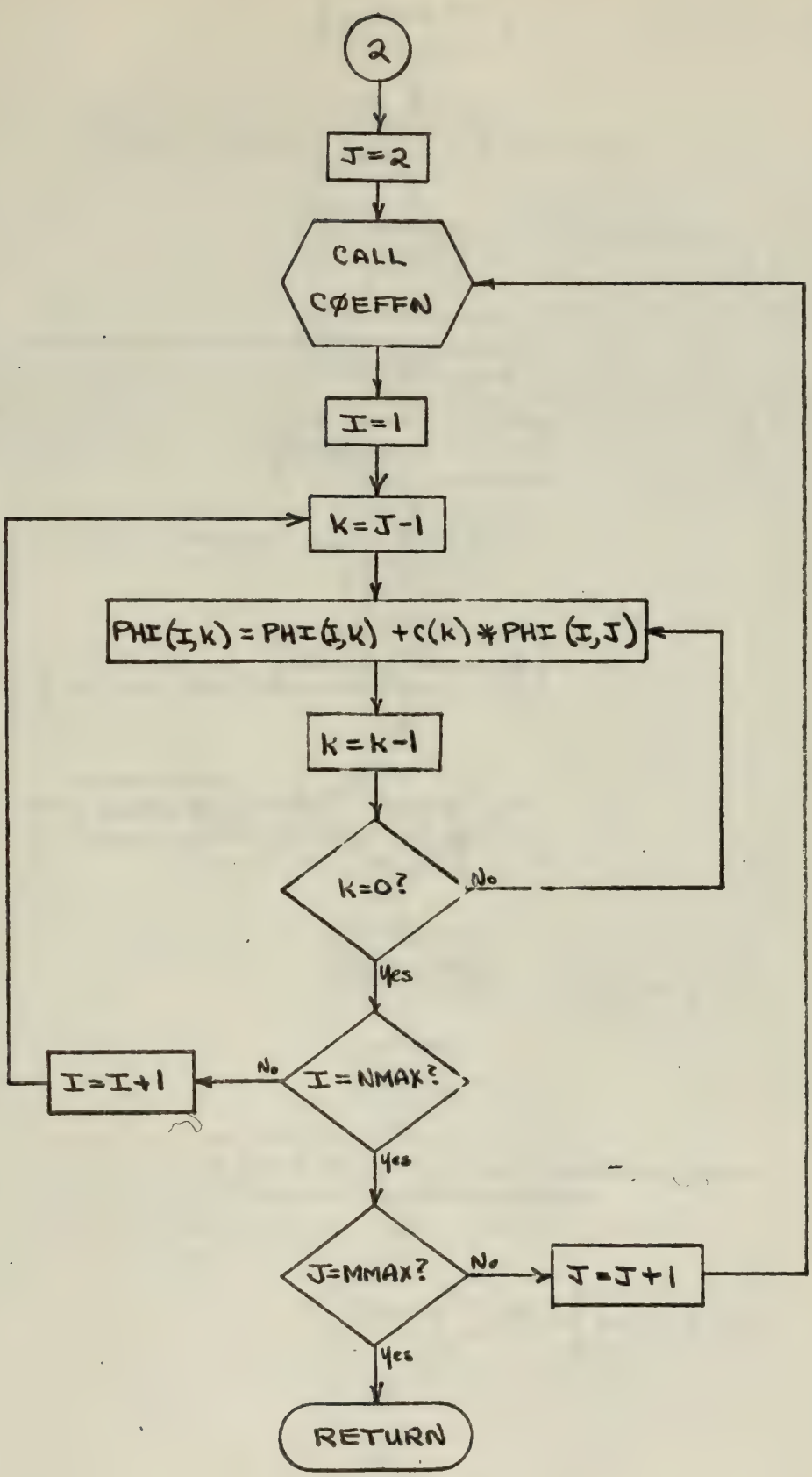

1
PHI(1,1) = PHI(1,1) + MEANZ
PRINT COEFFICIENTS OF APPROXIMATING POLYNOMIAL
PRINT
MEAN SQUARED
COMPUTED MEAN SQUARED
MAXIMUM DIFFERENCES
RETURN

# SUBRØUTINE PØLYXY

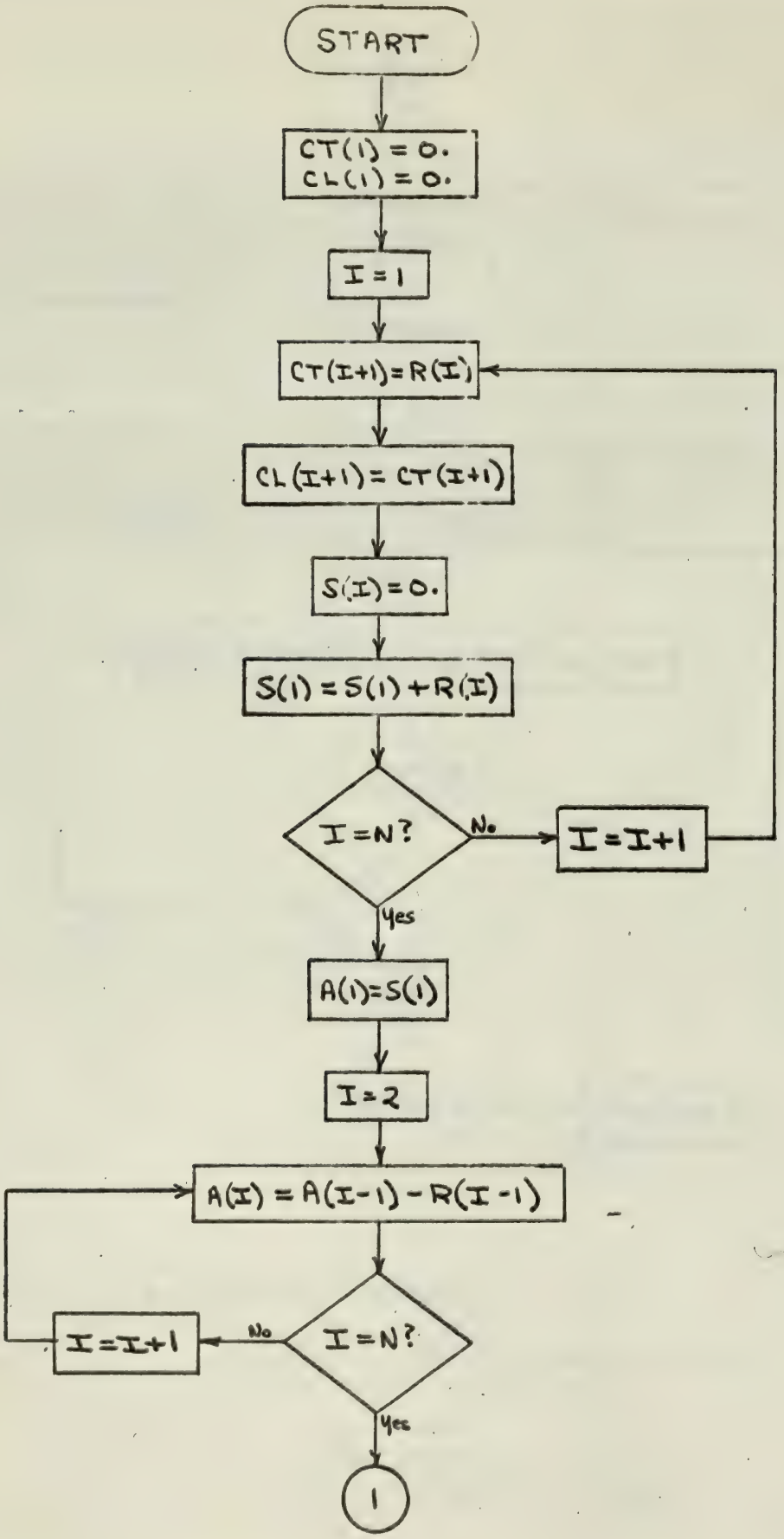


S/R
POLYXY

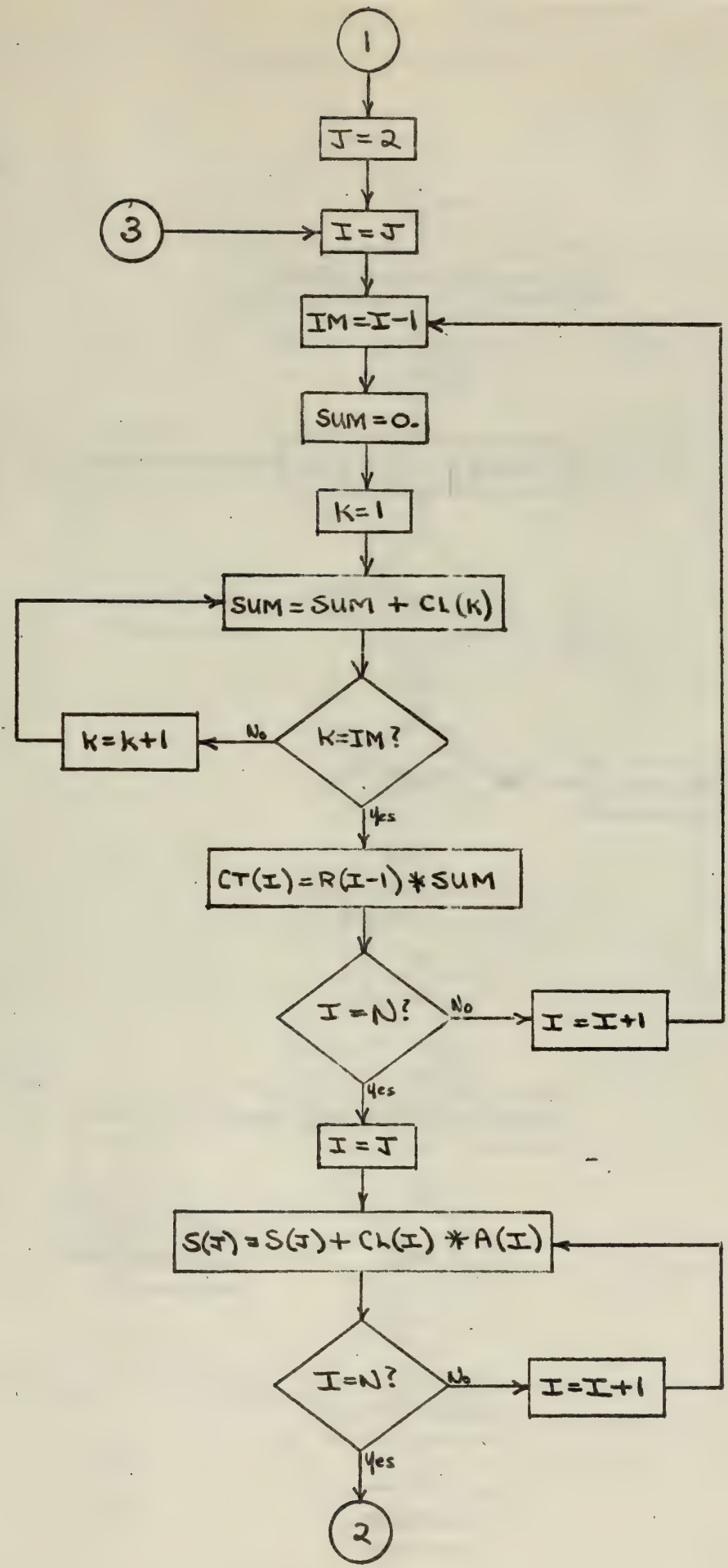

1
I=2
CALL
COEFFN
J=1
K=I-1
PHI(K,J)=PHI(K,J)+C(K)*PHI(I,J)
K=K-1
K=0?
No
Yes
J=JMAX?
No
J=J+1
Yes
I=NMAX?
No
I=I+1
Yes
I=1
R(I)=-YMEAN
I=MM1?
No
I=I+1
Yes
2


2
J=2
CALL
CØEFFN
I=1
K=J-1
PHI(I,K) = PHI(I,K) + C(K) * PHI (I,J)
K=K-1
K=0?
No
Yes
I=NMAX?
No
I=I+1
Yes
J=MMAX?
No
J=J+1
Yes
RETURN

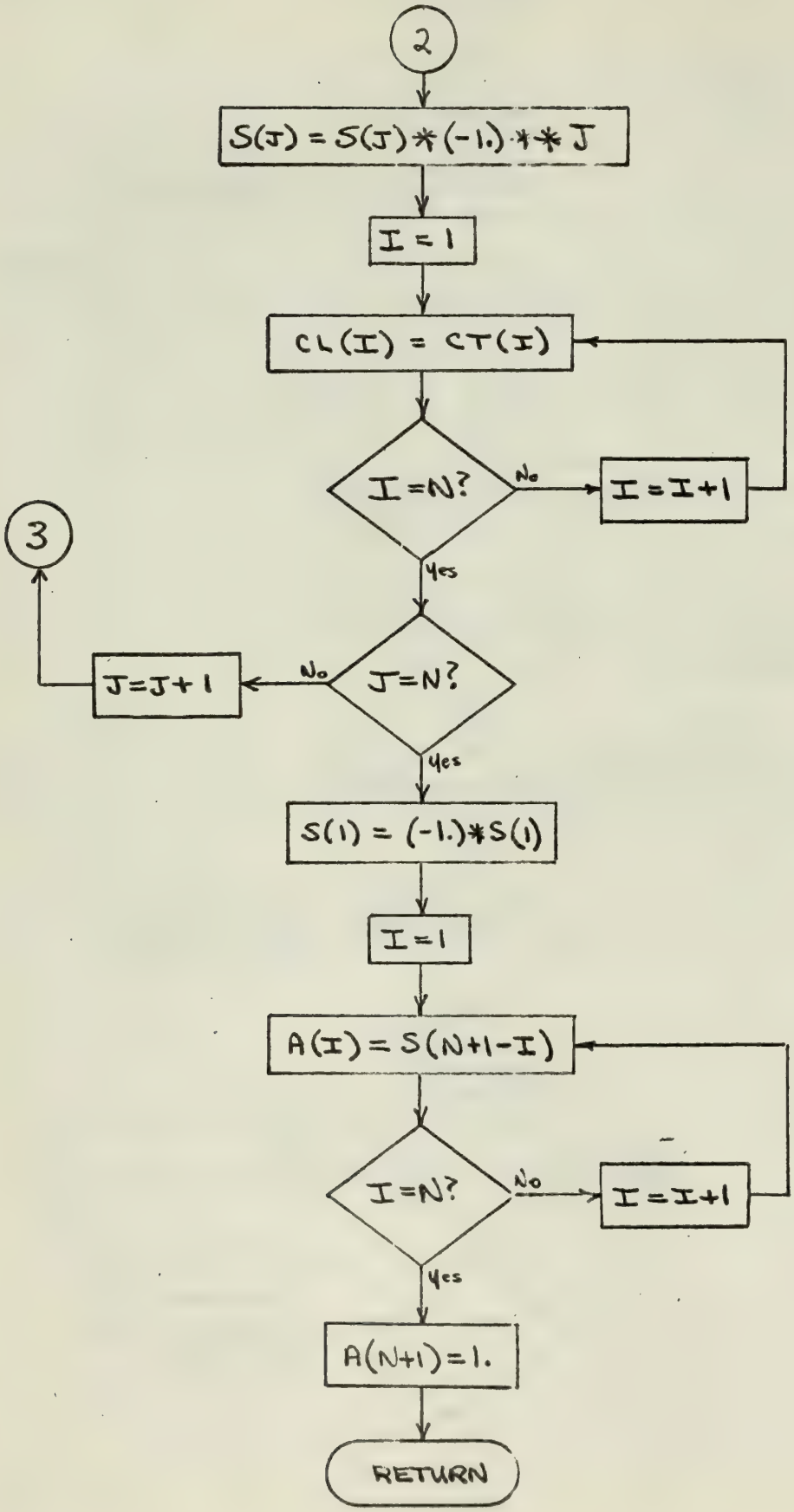
# SUBROUTINE COEFFN



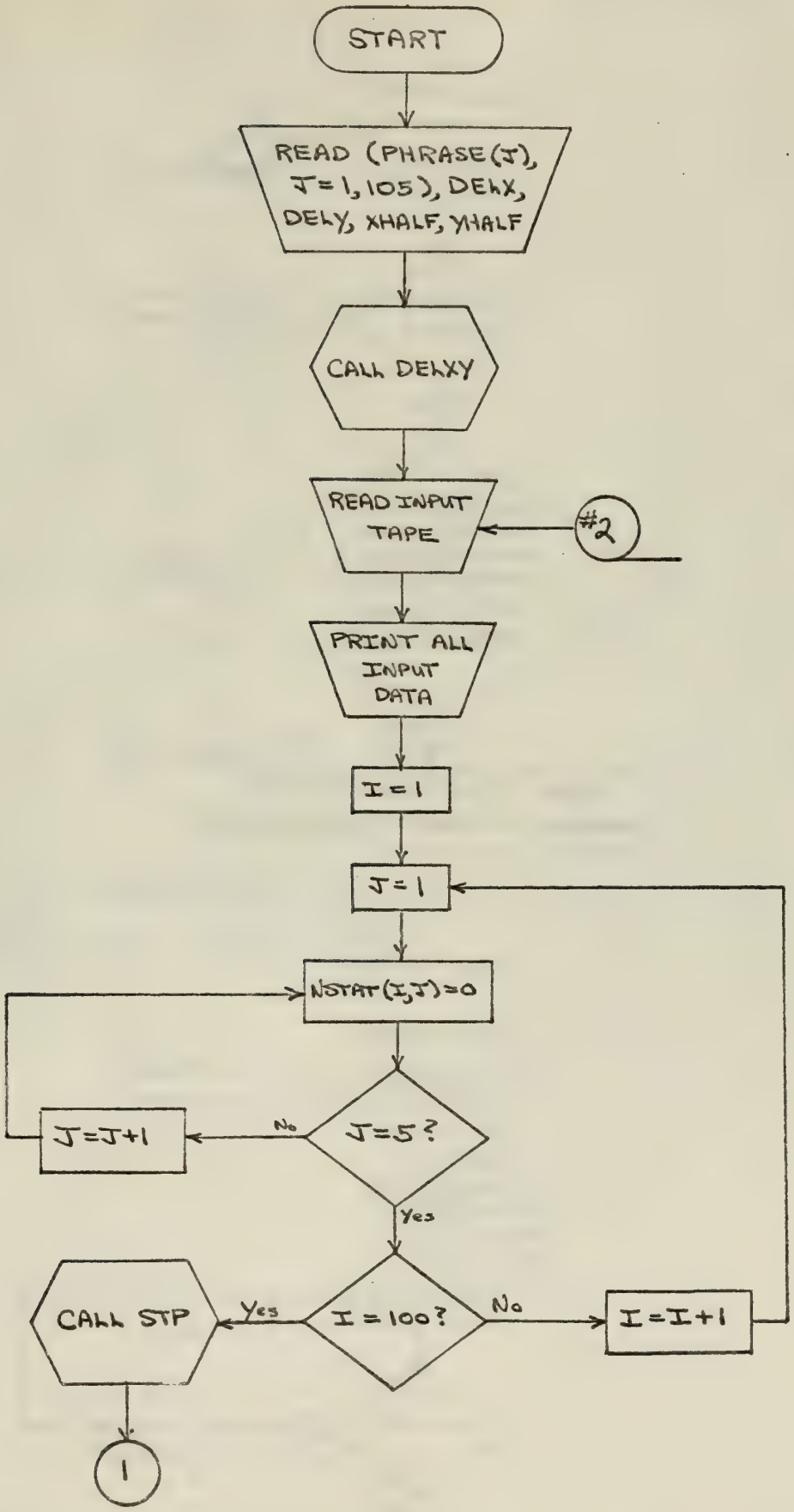
S/R COEFFN


1
J=2
3
I=J
IM=I-1
SUM=0.
K=1
SUM=SUM+CL(K)
K=IM?
No
K=K+1
Yes
CT(I)=R(I-1)*SUM
I=N?
No
I=I+1
Yes
I=J
S(J)=S(J)+CL(I)*A(I)
I=N?
No
I=I+1
Yes
2

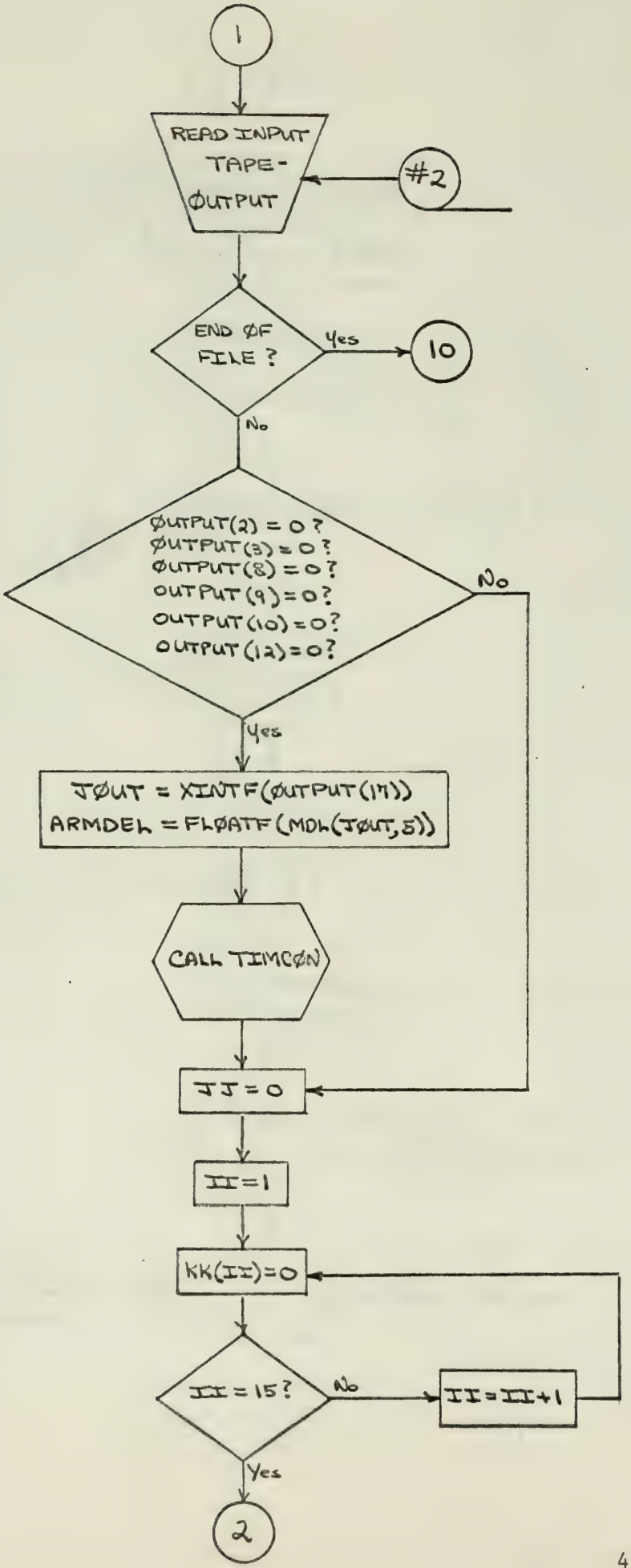
S/R CØEFFN

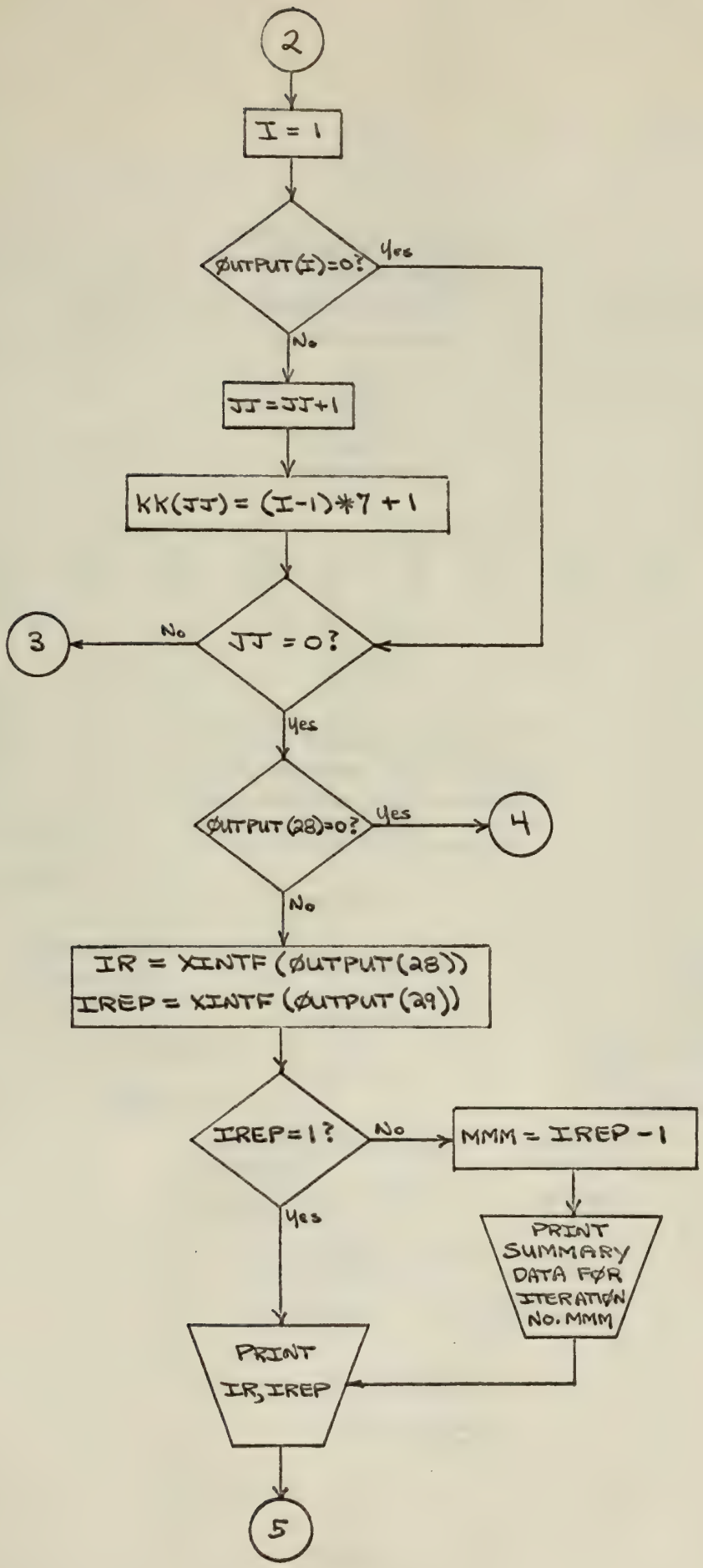

2
S(J) = S(J)*(-1.)**J
I = 1
CL(I) = CT(I)
I=N?
No
I = I+1
Yes
3
J=J+1
No
J=N?
Yes
S(1) = (-1.)*S(1)
I=1
A(I) = S(N+1-I)
I=N?
No
I = I+1
Yes
A(N+1) = 1.
RETURN

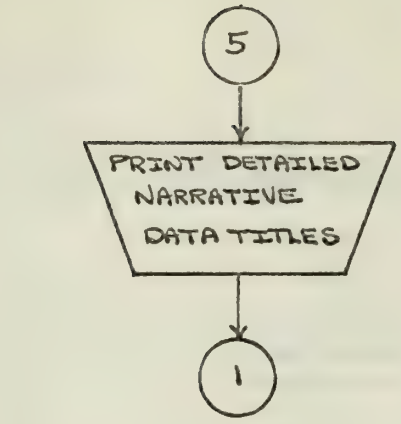
PROGRAM MINDEL1

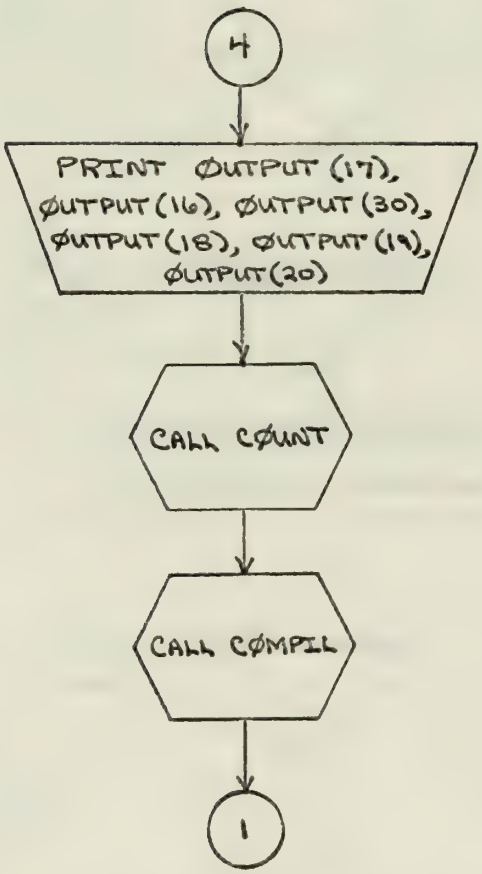


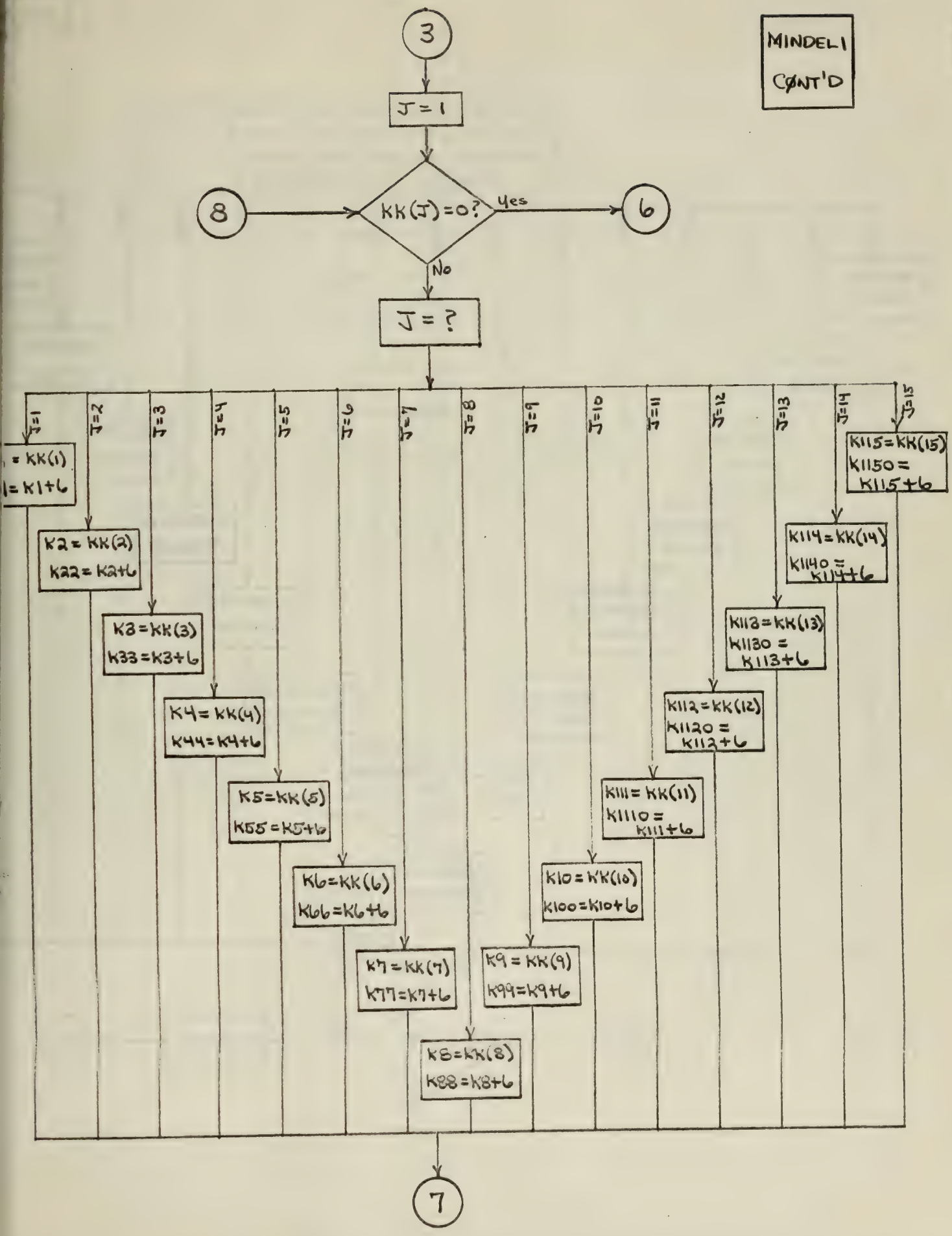
MINDEL1
CONT'D

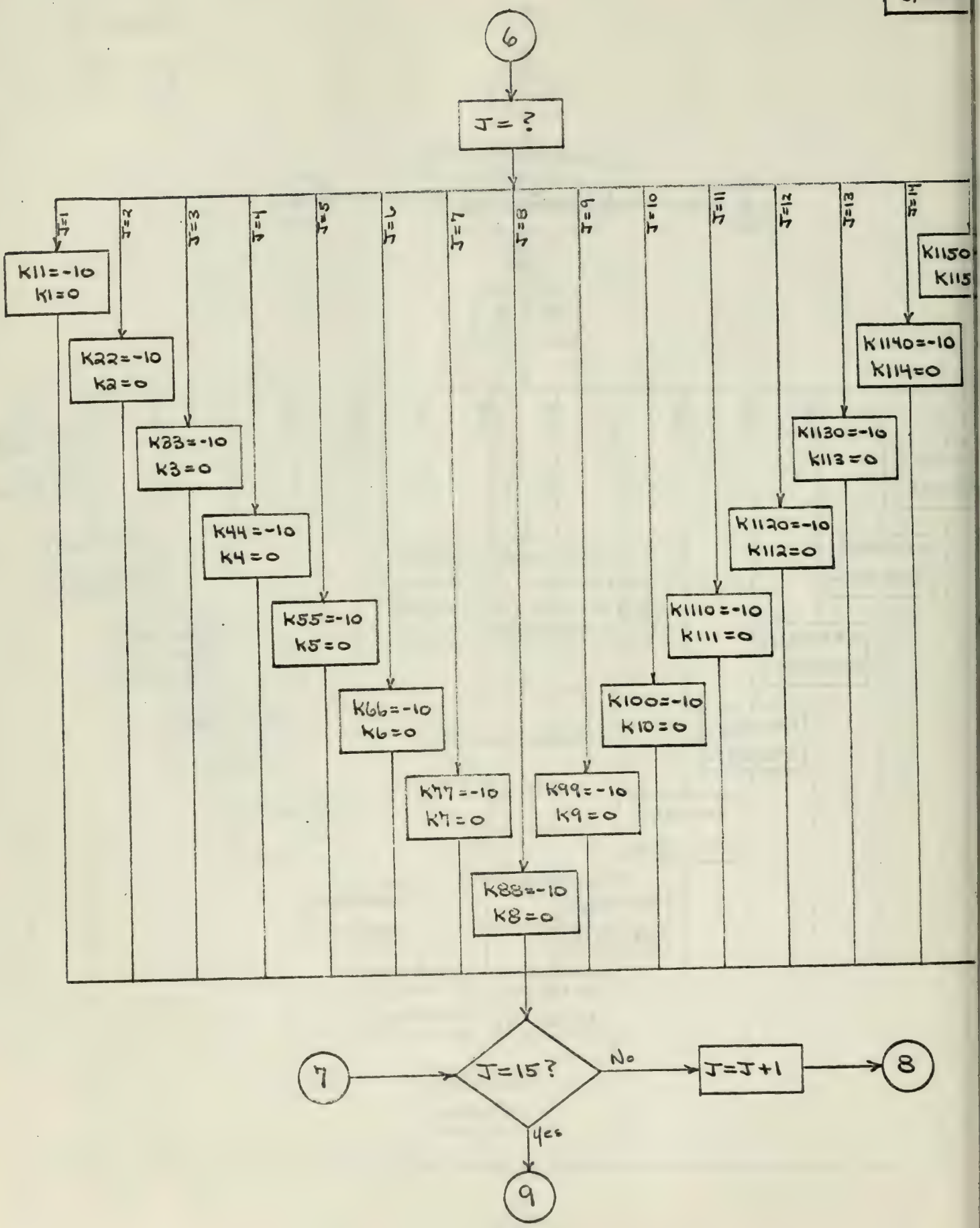



MINDEL1
CONT'D
2
I = 1
ØUTPUT(I)=0?
Yes
No
JJ=JJ+1
KK(JJ) = (I-1)*7 + 1
3
No
JJ = 0?
Yes
ØUTPUT(28)=0?
Yes
4
No
IR = XINTF (ØUTPUT(28))
IREP = XINTF (ØUTPUT(29))
IREP=1?
No
MMM = IREP - 1
Yes
PRINT SUMMARY DATA FØR ITERATIØN NO. MMM
PRINT IR, IREP
5

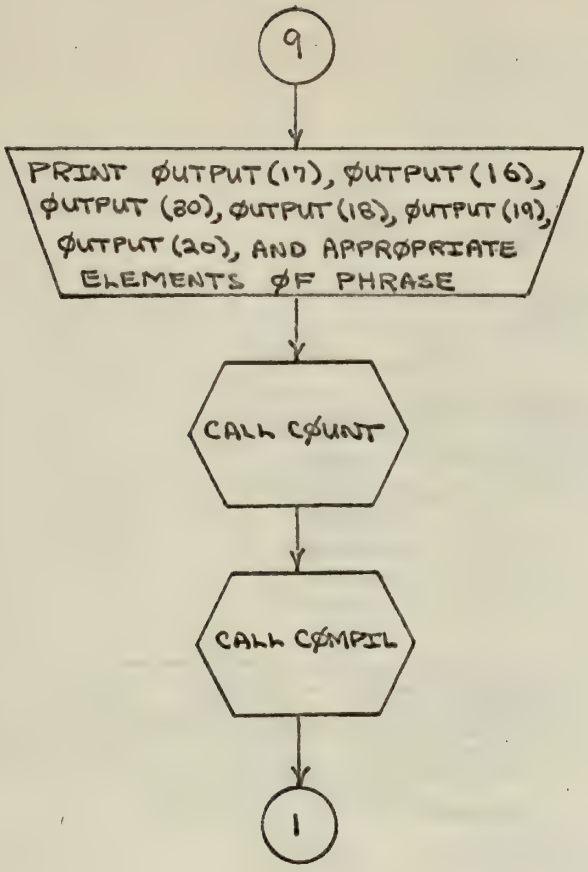

5
PRINT DETAILED
NARRATIVE
DATA TITLES
1

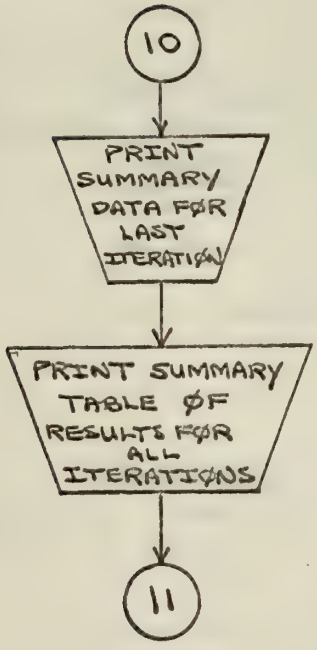

4
PRINT ØUTPUT (17),
ØUTPUT (16), ØUTPUT (30),
ØUTPUT (18), ØUTPUT (19),
ØUTPUT (20)
CALL CØUNT
CALL CØMPIL
1

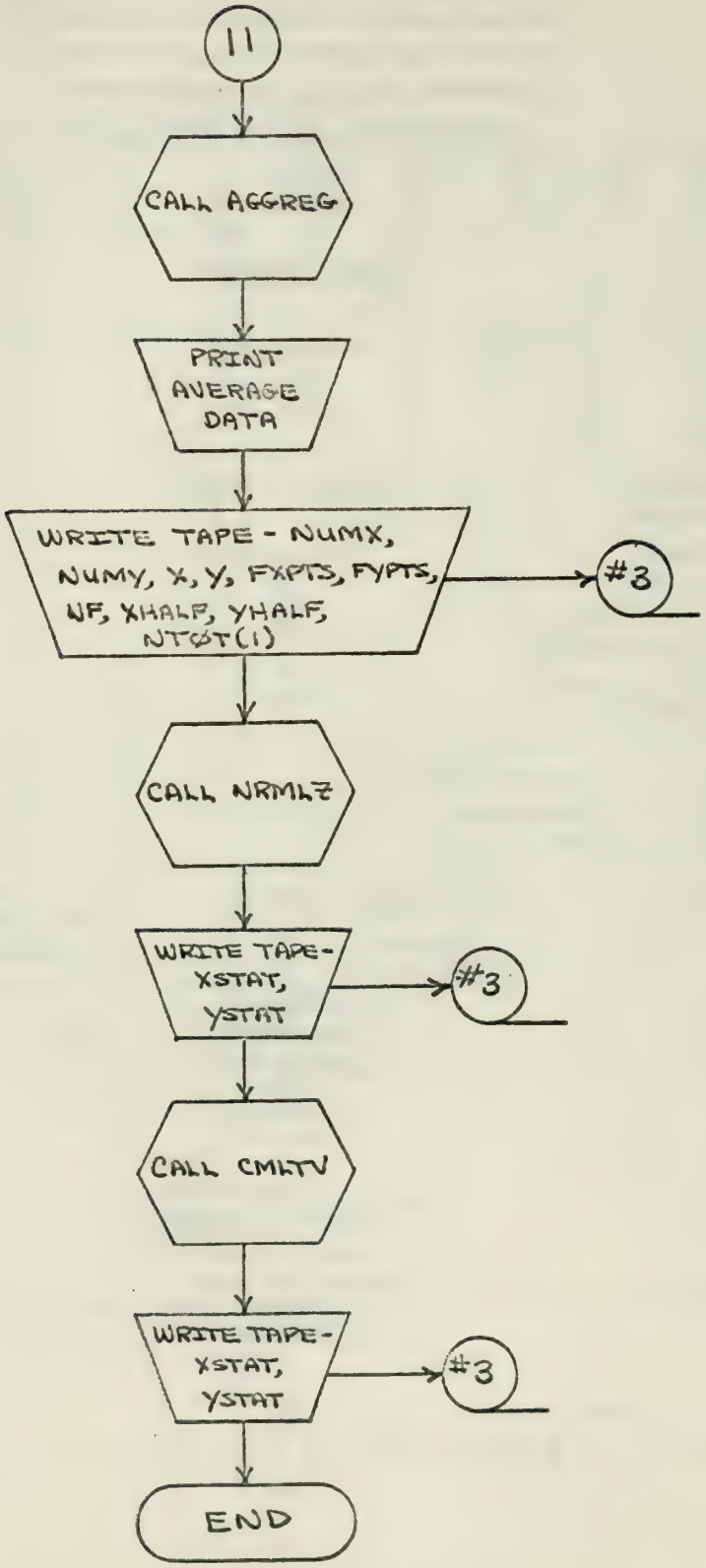

MINDEL1
CONT'D
3
J = 1
8
KK(J) = 0?
Yes
6
No
J = ?
J=1
J=2
J=3
J=4
J=5
J=6
J=7
J=8
J=9
J=10
J=11
J=12
J=13
J=14
J=15
= KK(1)
1 = K1+6
K2 = KK(2)
K22 = K2+6
K3 = KK(3)
K33 = K3+6
K4 = KK(4)
K44 = K4+6
K5 = KK(5)
K55 = K5+6
K6 = KK(6)
K66 = K6+6
K7 = KK(7)
K77 = K7+6
K8 = KK(8)
K88 = K8+6
K9 = KK(9)
K99 = K9+6
K10 = KK(10)
K100 = K10+6
K111 = KK(11)
K1110 = K111+6
K112 = KK(12)
K1120 = K112+6
K113 = KK(13)
K1130 = K113+6
K114 = KK(14)
K1140 = K114+6
K115 = KK(15)
K1150 = K115+6
7


6
J = ?
J=1
J=2
J=3
J=4
J=5
J=6
J=7
J=8
J=9
J=10
J=11
J=12
J=13
J=14
K11=-10
K1=0
K22=-10
K2=0
K33=-10
K3=0
K44=-10
K4=0
K55=-10
K5=0
K66=-10
K6=0
K77=-10
K7=0
K88=-10
K8=0
K99=-10
K9=0
K100=-10
K10=0
K1110=-10
K111=0
K1120=-10
K112=0
K1130=-10
K113=0
K1140=-10
K114=0
K1150=
K115
7
J=15?
No
J=J+1
8
Yes
9

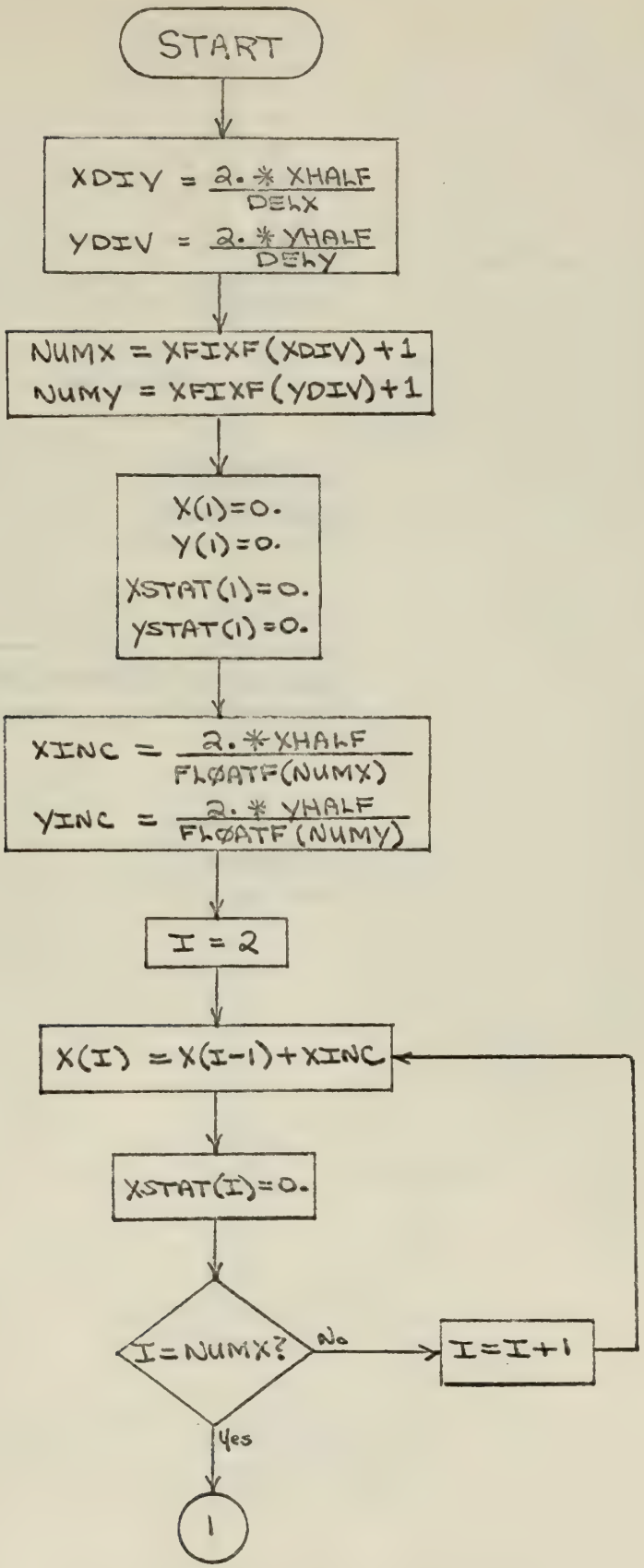
MINDEL1
CØNT'D

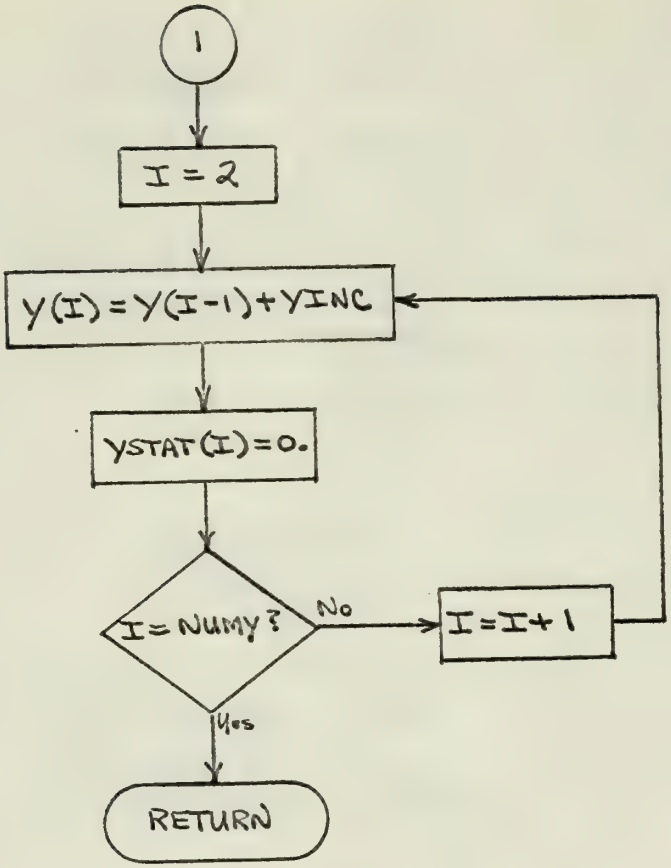


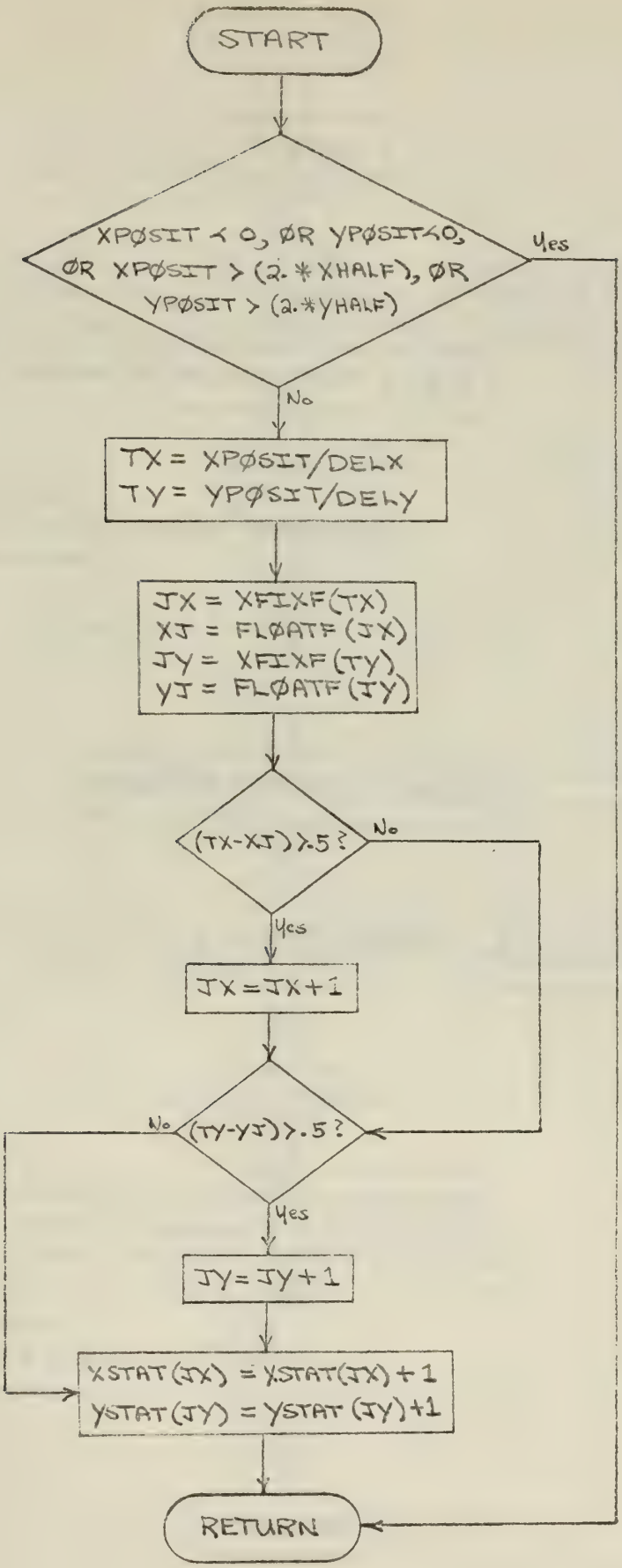

11
CALL AGGREG
PRINT AVERAGE DATA
WRITE TAPE - NUMX, NUMY, X, Y, FXPTS, FYPTS, NF, XHALF, YHALF, NTOT(1)
#3
CALL NRMLZ
WRITE TAPE- XSTAT, YSTAT
#3
CALL CMLTV
WRITE TAPE- XSTAT, YSTAT
#3
END

# SUBROUTINE DELXY

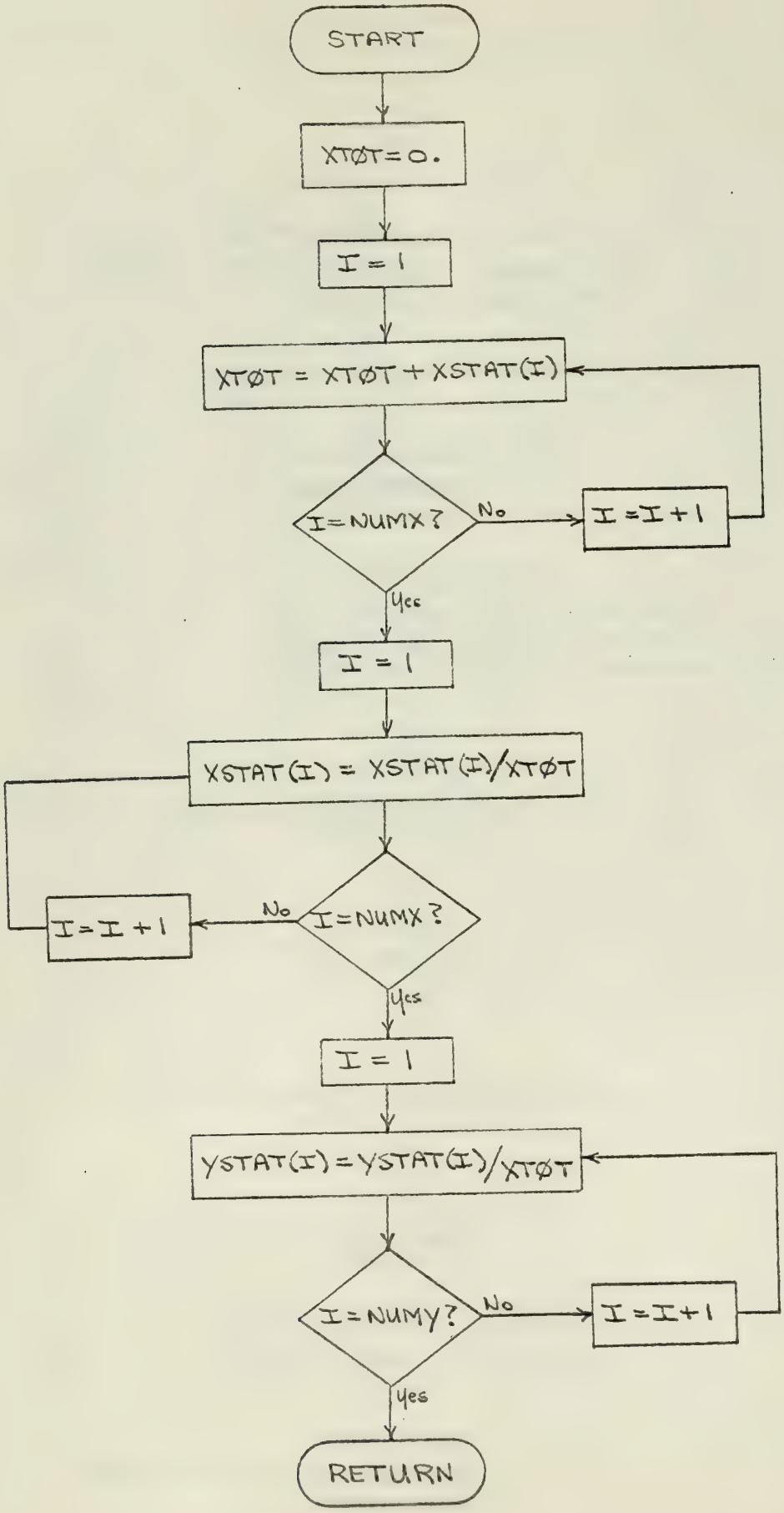



1
I = 2
Y(I) = Y(I-1) + YINC
YSTAT(I) = 0.
I = NUMY ?
No
I = I + 1
Yes
RETURN

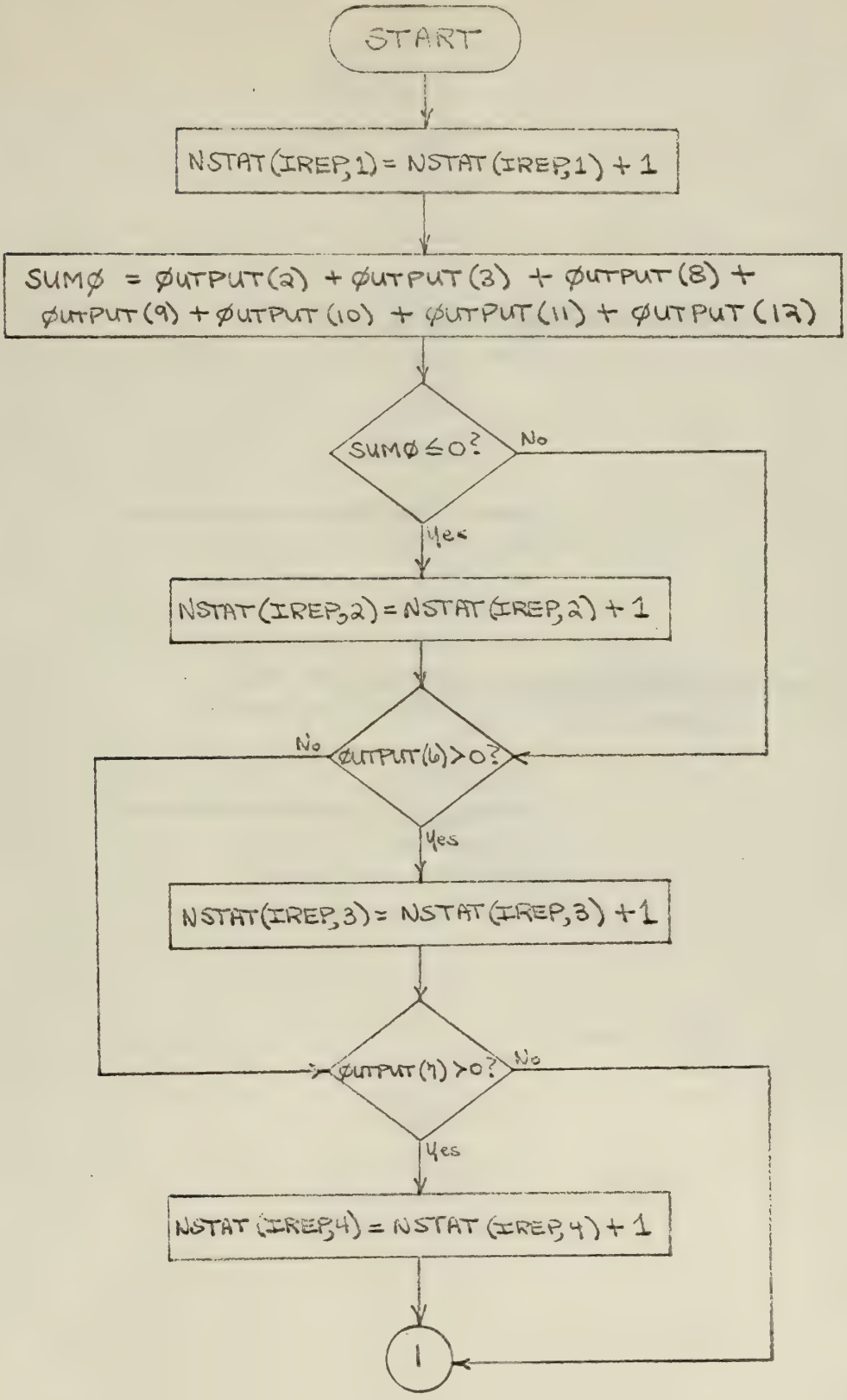
# SUBROUTINE COUNT

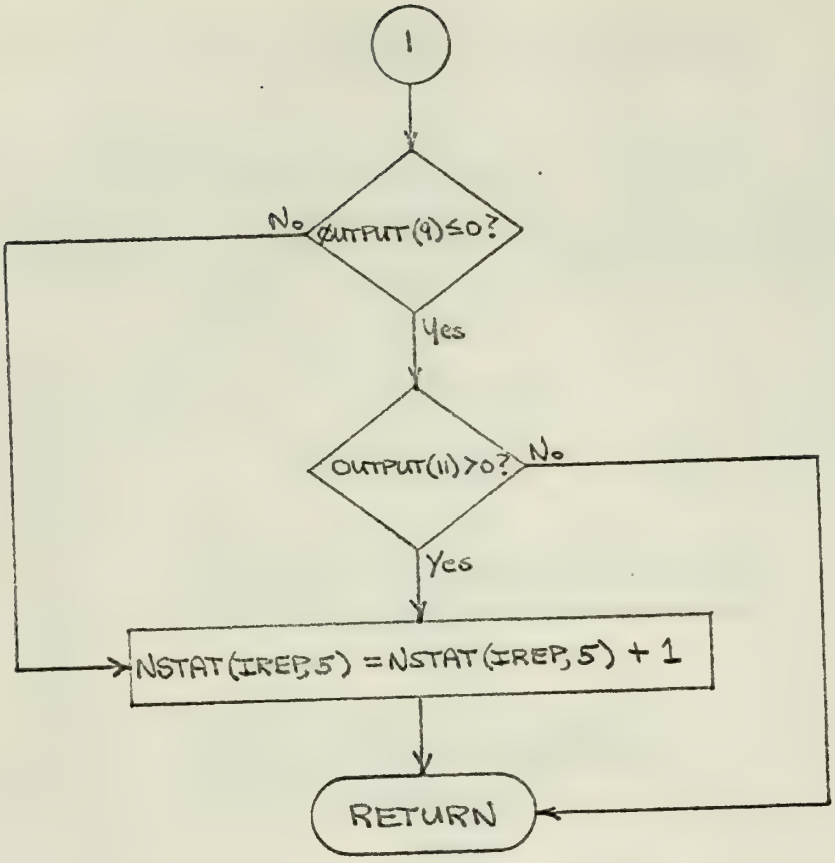


# SUBROUTINE NRMLZ

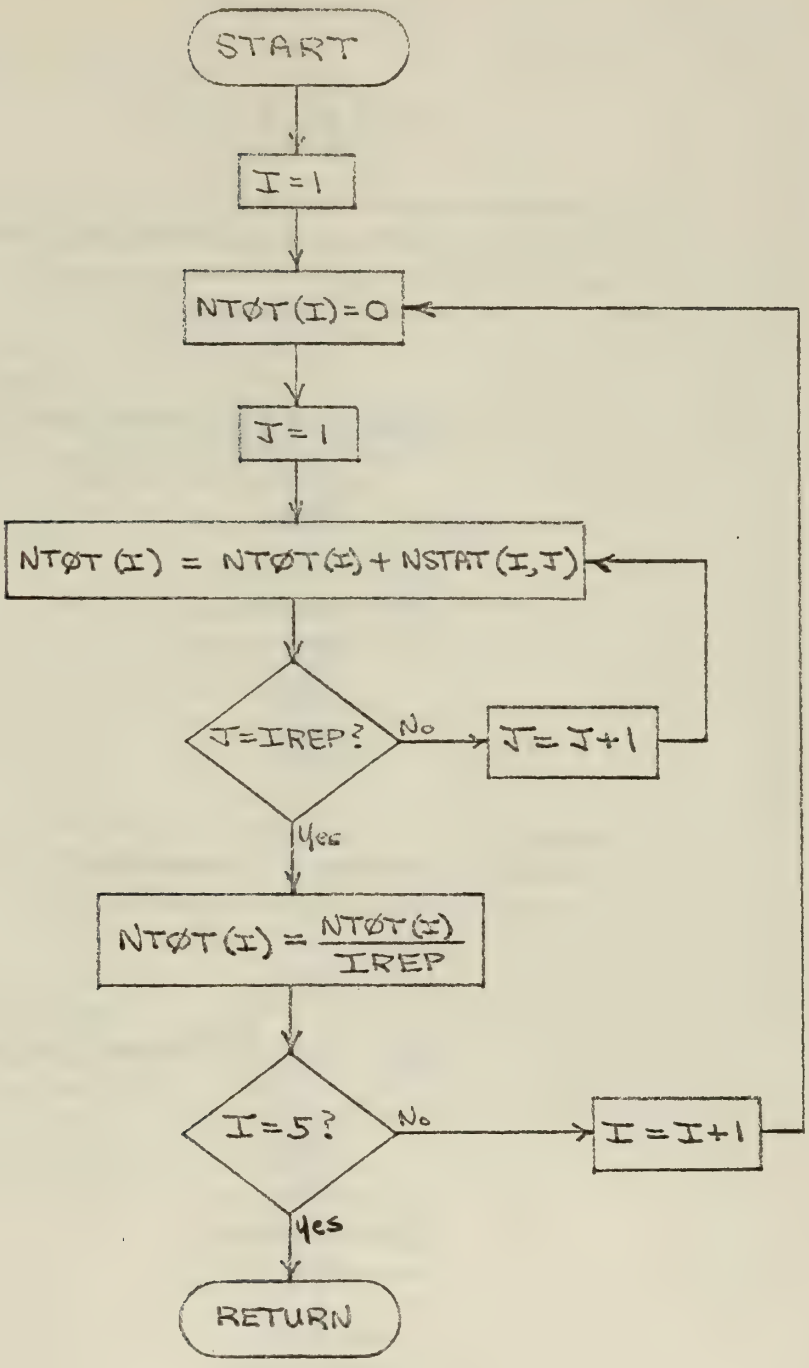


# SUBROUTINE COMPIL

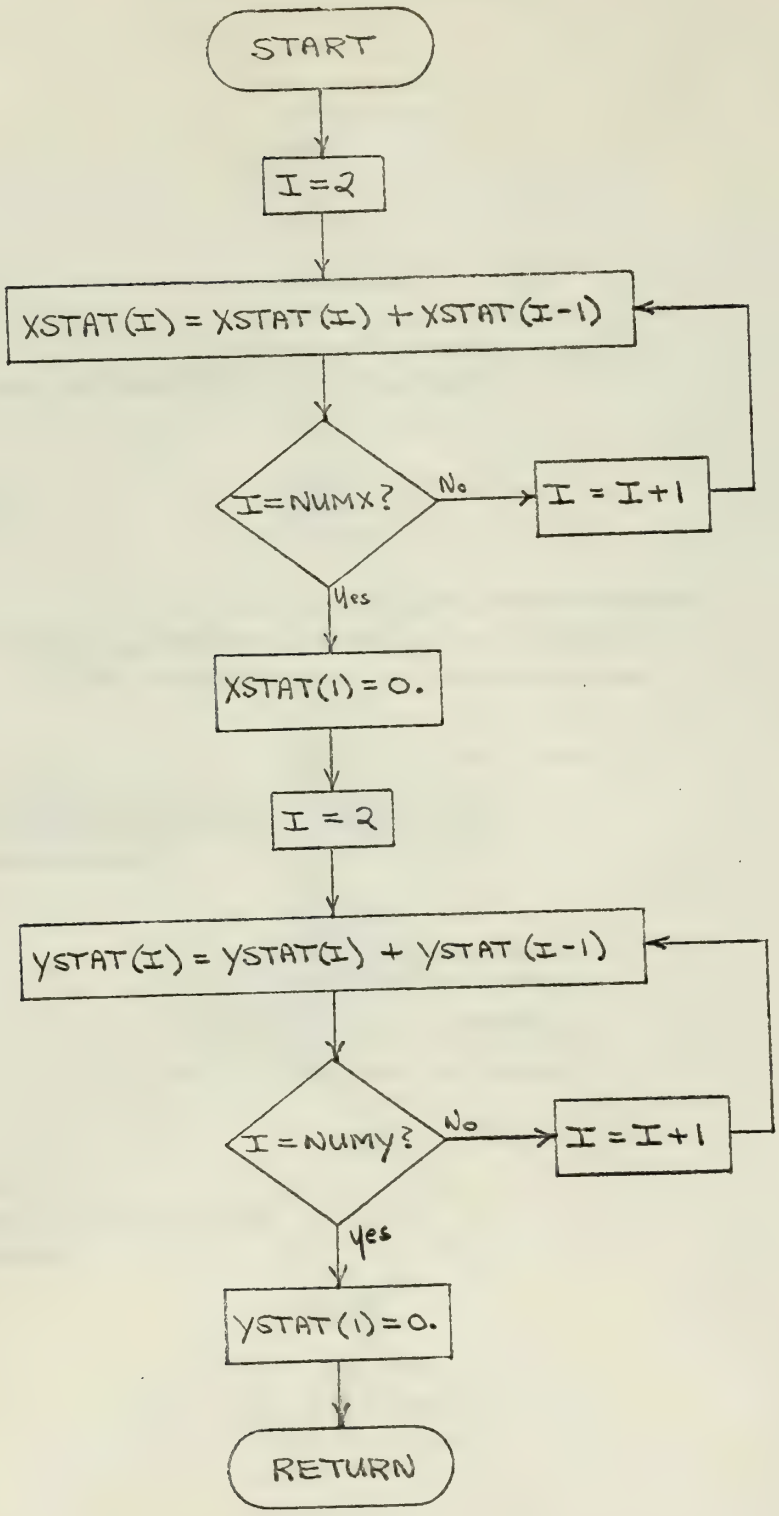



1
No
OUTPUT(9)≤0?
Yes
OUTPUT(11)>0?
No
Yes
NSTAT(IREP,5) = NSTAT(IREP,5) + 1
RETURN

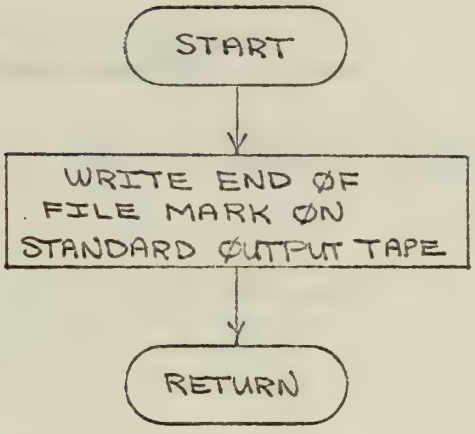
# SUBROUTINE AGGREG



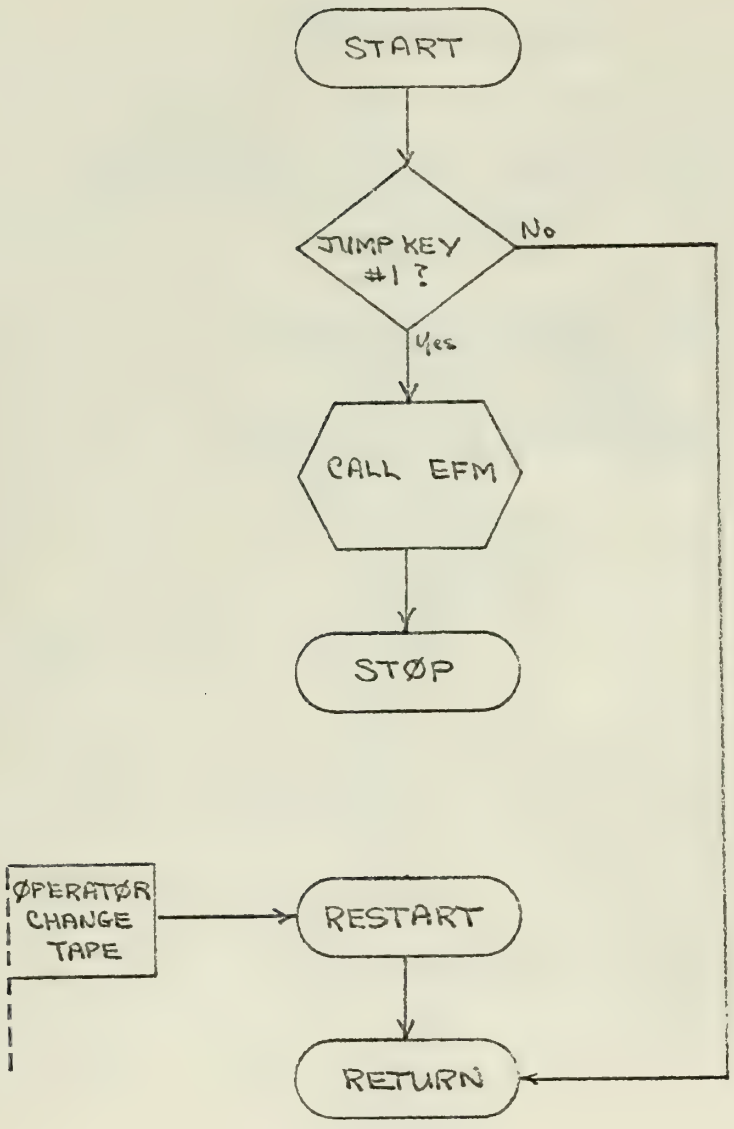
SUBROUTINE CMLTV



## SUBROUTINE EFM



# Subroutine Stp



INITIAL DISTRIBUTION LIST

| | | No. Copies |
|---|---|---|
| 1. | Defense Documentation Center<br>Cameron Station<br>Alexandria, Virginia 22314 | 20 |
| 2. | Library<br>U. S. Naval Postgraduate School<br>Monterey, California 93940 | 2 |
| 3. | Office of the Assistant to the Chief of Naval Operations for War Gaming Matters (OP-06C)<br>Department of the Navy<br>Washington, D. C. 20350 | 2 |
| 4. | Prof. A. F. Andrus (Thesis Advisor)<br>Department of Operations Analysis<br>U. S. Naval Postgraduate School<br>Monterey, California 93940 | 1 |
| 5. | Mr. Vern Patton<br>Naval Weapons Laboratory (Code KW)<br>Dahlgren, Virginia 22448 | 1 |
| 6. | U. S. Naval Mine Warfare School<br>U. S. Naval Base<br>Charleston, South Carolina 29408 | 1 |
| 7. | LT T. G. Krumm, USN<br>USS Cowell (DD-547)<br>c/o FPO, San Francisco, California 96601 | 1 |
| 8. | LT S. V. Snyder, USN<br>Strategic Analysis Support Group<br>Johns Hopkins University Applied Physics Lab<br>Silver Spring, Maryland 20910 | 1 |

**DOCUMENT CONTROL DATA - R&D**

*(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)*

| 1. ORIGINATING ACTIVITY (Corporate author) | 2a. REPORT SECURITY CLASSIFICATION |
|---|---|
| U. S. Naval Postgraduate School<br>Monterey, California 93940 | Unclassified |
| | 2b. GROUP NA |

3. REPORT TITLE

MINE DELIVERY MODEL: A COMPUTER SIMULATION

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

- - -

5. AUTHOR(S) (Last name, first name, initial)

Krumm, Theodore G., Lieutenant, U. S. Navy
Snyder, Stephen V., Lieutenant, U. S. Navy

| 6. REPORT DATE | 7a. TOTAL NO. OF PAGES | 7b. NO. OF REFS |
|---|---|---|
| May 1966 | 422 | 17 |

| 8a. CONTRACT OR GRANT NO. | 9a. ORIGINATOR'S REPORT NUMBER(S) |
|---|---|
| b. PROJECT NO. - - - | |
| c. | 9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report) - - - |
| d. | |

10. AVAILABILITY/LIMITATION NOTICES

Approved for Public Release; Distribution Unlimited

| 11. SUPPLEMENTARY NOTES | 12. SPONSORING MILITARY ACTIVITY |
|---|---|
| NA | U. S. Naval Postgraduate School<br>Monterey, California 93940 |

13. ABSTRACT

The Mine Delivery Model is a Monte Carlo computer simulation of the delivery of mines in detail by ships, submarines, and/or aircraft. Inputs are of such a nature as to permit the simulated execution of any specific mining plan and the outputs provide sufficient information for subsequent evaluation of the plan by a threat assessment model.

The model as programmed for the CDC 1604 Computer consists of five programs. The first two programs convert input data, the third and main program is the mine delivery simulation, the fourth program provides a detailed printout of both inputs and game results, i.e., the calculated parameters for each mine layed accompanied by selected statistics, and the fifth program provides statistics and graphs on the distributions of the mine positions.

DD FORM 1 JAN 64 1473

| 14. KEY WORDS | LINK A | | LINK B | | LINK C | |
|---|---|---|---|---|---|---|
| | ROLE | WT | ROLE | WT | ROLE | WT |
| Mine Delivery<br>Mine Warfare<br>War Gaming, Digital Computer<br>Simulation, Digital Computer | | | | | | |

## INSTRUCTIONS

1. ORIGINATING ACTIVITY: Enter the name and address of the contractor, subcontractor, grantee, Department of Defense activity or other organization *(corporate author)* issuing the report.

2*a*. REPORT SECURITY CLASSIFICATION: Enter the overall security classification of the report. Indicate whether "Restricted Data" is included. Marking is to be in accordance with appropriate security regulations.

2*b*. GROUP: Automatic downgrading is specified in DoD Directive 5200.10 and Armed Forces Industrial Manual. Enter the group number. Also, when applicable, show that optional markings have been used for Group 3 and Group 4 as authorized.

3. REPORT TITLE: Enter the complete report title in all capital letters. Titles in all cases should be unclassified. If a meaningful title cannot be selected without classification, show title classification in all capitals in parenthesis immediately following the title.

4. DESCRIPTIVE NOTES: If appropriate, enter the type of report, e.g., interim, progress, summary, annual, or final. Give the inclusive dates when a specific reporting period is covered.

5. AUTHOR(S): Enter the name(s) of author(s) as shown on or in the report. Enter last name, first name, middle initial. If military, show rank and branch of service. The name of the principal author is an absolute minimum requirement.

6. REPORT DATE: Enter the date of the report as day, month, year; or month, year. If more than one date appears on the report, use date of publication.

7*a*. TOTAL NUMBER OF PAGES: The total page count should follow normal pagination procedures, i.e., enter the number of pages containing information.

7*b*. NUMBER OF REFERENCES: Enter the total number of references cited in the report.

8*a*. CONTRACT OR GRANT NUMBER: If appropriate, enter the applicable number of the contract or grant under which the report was written.

8*b*, 8*c*, & 8*d*. PROJECT NUMBER: Enter the appropriate military department identification, such as project number, subproject number, system numbers, task number, etc.

9*a*. ORIGINATOR'S REPORT NUMBER(S): Enter the official report number by which the document will be identified and controlled by the originating activity. This number must be unique to this report.

9*b*. OTHER REPORT NUMBER(S): If the report has been assigned any other report numbers *(either by the originator or by the sponsor)*, also enter this number(s).

10. AVAILABILITY/LIMITATION NOTICES: Enter any limitations on further dissemination of the report, other than those imposed by security classification, using standard statements such as:

(1) "Qualified requesters may obtain copies of this report from DDC."

(2) "Foreign announcement and dissemination of this report by DDC is not authorized."

(3) "U. S. Government agencies may obtain copies of this report directly from DDC. Other qualified DDC users shall request through _____."

(4) "U. S. military agencies may obtain copies of this report directly from DDC. Other qualified users shall request through _____."

(5) "All distribution of this report is controlled. Qualified DDC users shall request through _____."

If the report has been furnished to the Office of Technical Services, Department of Commerce, for sale to the public, indicate this fact and enter the price, if known.

11. SUPPLEMENTARY NOTES: Use for additional explanatory notes.

12. SPONSORING MILITARY ACTIVITY: Enter the name of the departmental project office or laboratory sponsoring *(paying for)* the research and development. Include address.

13. ABSTRACT: Enter an abstract giving a brief and factual summary of the document indicative of the report, even though it may also appear elsewhere in the body of the technical report. If additional space is required, a continuation sheet shall be attached.

It is highly desirable that the abstract of classified reports be unclassified. Each paragraph of the abstract shall end with an indication of the military security classification of the information in the paragraph, represented as *(TS), (S), (C), or (U)*.

There is no limitation on the length of the abstract. However, the suggested length is from 150 to 225 words.

14. KEY WORDS: Key words are technically meaningful terms or short phrases that characterize a report and may be used as index entries for cataloging the report. Key words must be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location, may be used as key words but will be followed by an indication of technical context. The assignment of links, roles, and weights is optional.

thesK875
Mine delivery model :
3 2768 002 11550 3
DUDLEY KNOX LIBRARY c.1